ISSN : 2527-5836
e-ISSN : 2528-0074

Vol. 4 No. 3, Januari 2020

JISKa

Jurnal Informatika Sunan Kalijaga

Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
UIN Sunan Kalijaga Yogyakarta

PENGELOLA JISKa Edisi Januari 2020

ISSN : 2527-5836
e-ISSN : 2528-0074

JISKa
Vol. 4, No. 3, JANUARI 2020

DAFTAR ISI

Penerapan Algoritma Hill Cipher Dan Least Significant Bit (LSB) Untuk Pengamanan Pesan Pada Citra Digital **138-148**
Desimeri Laoli
Bosker Sinaga
Anita Sindar R M Sinaga

Pengembangan Aplikasi Computer Based Test dengan Protokol Two Central Facilities **149-155**
Muhammad Nasyithul Ibad
Syarif Alqoroni
Muhammad Ammarullah Ridho
Khadijah Fahmi Hayati Holle

Deteksi Serangan Distributed Denia of Services (DDOS) Berbasis HTTP Menggunakan Metode Fuzzy Sugeno **156-164**
Nadila Sugianti
Yayang Galuh
Salma Fatia
Khadijah Fahmi Hayati Holle

DESAIN DAN IMPLEMENTASI SIMULASI INTRUSION INDEX BERBASIS SISTEM PAKAR DENGAN METODE FORWARD CHAINING **165-172**
Mardian Mardian
H. Jemakmun
Linda Atika

173-184
Pengembangan Sistem Pemetaan Status Mutu Air Sungai Berbasis Web Menggunakan Extreme Programming
Shofwatul ‘Uyun
Ramadhan Salahudin Al Ayubi
Yulia Siti Ambarwati

Journal Classification Based on Abstract Using Cosine Similarity and Support Vector Machine **185-192**
Muhammad Habibi
Puji Winar Cahyo

Diagnosa Penyakit Demam Berdarah Dengue (DBD) menggunakan Metode Learning Vector Quantization (LVQ) **193-201**
Firman Tawakal
Ahmedika Azkiya

JISKa, Vol. 4, No. 3, JANUARI, 2020, Pp. 138 – 148
ISSN : 2527 – 5836 (print) | 2528 – 0074 (online)

Penerapan Algoritma Hill Cipher Dan Least Significant Bit (LSB) Untuk Pengamanan Pesan Pada Citra Digital

Desimeri Laoli[3], Bosker Sinaga[2], Anita Sindar[3]
Teknik Informatika STMIK Pelita Nusantara
Jl. Iskandar Muda No. 1 Medan
Email : desimeri9094laoli@gmail.com[1], sinagab8@gmail.com[2], haito_ita@yahoo.com[3]

Abstract

Nowadays people exchange information in digital media such as text, audio, video and imagery. The development of Information and Communication makes the delivery of information and data more efficient. Current developments in technology which are very significant have an impact on the community in exchanging information and communicating. Confidential hidden data can also be in the form of image, audio, text, or video. The Hill Chiper algorithm uses a matrix of size m x m as a key for encryption and decryption. One way to recover the original text is of course to guess the decryption key, so the process of guessing the decryption key must be difficult. break ciphertext into palintext without knowing which key to use. The LSB part that is converted to the value of the message to be inserted. After affixing a secret message, each pixel is rebuilt into a whole image that resembles the original image media. The Hill Cipher algorithm is used to determine the position of the plaintext encryption into a random ciphertext. 2. Testing text messages using the hill cipher algorithm successfully carried out in accordance with the flow or the steps so as to produce a ciphertext in the form of randomization of the letters of the alphabet.

Keywords : Image, Confidential Data, Cryptography, Hill Chiper Algorithm, LSB

Abstrak

Saat ini masyarakat bertukar informasi dalam media digital seperti teks, audio, video, dan citra. Perkembangan Informasi dan Komunikasi menjadikan kegiatan penyampaian informasi maupun data menjadi lebih efesien. Perkembangan teknologi saat ini yang sangat signifikan memberikan dampak bagi masyarakat dalam bertukar informasi maupun melakukan komunikasi. Data rahasia yang disembunyikan juga dapat berupa citra, audio, teks, atau video. Algoritma Hill Chiper menggunakan matriks berukuran m x m sebagai kunci untuk melakukan enkripsi dan dekripsi. Satu cara untuk mendapatkan kembali naskah asli tentunya dengan menerka kunci dekripsi, jadi proses menerka kunci dekripsi harus menjadi sesuatu yang sulit. memecahkan chiperteks menjadi palinteks tanpa mengetahui kunci yang digunakan. Bagian LSB yang diubah menjadi nilai dari pesan yang akan disisipkan. Setelah dibubuhi pesan rahasia, setiap pixel dibangun kembali menjadi gambar yang utuh menyerupai dengan media gambar semula. Algoritma Hill Cipher yang digunakan untuk menentukan posisi enkrip plaintext menjadi sebuah ciphertext yang secara acak. 2. Pengujian pesan teks menggunakan algoritma hill cipher berhasil dilakukan sesuai tepat dengan alur atau langkah-langkahnya sehingga menghasilkan cipherteks yang berupa pengacakan huruf abjad.

Kata Kunci : Citra, Data Rahasia, Kriptografi, Algoritma Hill Chiper, LSB

1. PENDAHULUAN

Keamanan dan kerahasiaan informasi yakni tolak ukur yang sangat penting dalam sistem informasi. Meningkatnya kemajuan di bidang informasi dan teknologi yang menyebabkan adanya cara-cara terbaru, yang digunakan dengan tidak bertanggung jawab oleh beberapa oknum yang menyalahgunakan fungsi keamanan akan sebuah sistem informasi. Informasi tersebar ke tangan okum lain dapat menimbulkan efek negatif untuk pemilik informasi. Secara

umum informasi dikategorikan menjadi dua, yaitu informasi yang bersifat rahasia dan informasi yang tidak bersifat rahasia. Informasi yang tidak bersifat rahasia biasanya tidak akan terlalu diperhatikan. Informasi bersifat rahasia yaitu setiap informasi yang ada didalamnya sangat berharga bagi pihak yang membutuhkan karena informasi tersebut dapat dengan mudah digandakan. Informasi bisa dalam berbentuk sebuah file ataupun string [1]. Mengacu pada permasalahan yang dibahas maka diperlukan untuk merancang sebuah sistem keamanan yang dapat melindungi data yang dianggap penting dengan penyandian data, serta membuat kunci rahasia untuk dapat membuka data tersebut yang sulit untuk di deteksi oleh pihak yang tidak berhak. Gabungan dari metode kriptografi yakni algoritma Hill Cipher untuk enkripsi pesan dengan metode steganografi yaitu Least Significant Bit (LSB) dapat menambah keamanan dalam sebuah pesan [2]. Menurut Jurnal Fresly Nandar Pabokory, Kriptografi merupakan salah satu ilmu maupun seni untuk menjaga kerahasiaan sebuah pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya. Proses enkripsi dilakukan menggunakan suatu algoritma dengan beberapa alur kerja dari proses enkripsi. Algoritma yang dipakai pada penyusunan skripsi berikut ini yaitu Hill Cipher [3]. Jurnal Jane Irma Sari, Steganografi merupakan seni untuk menyembunyikan pesan di dalam media digital sedemikian rupa sehingga orang lain tidak menyadari ada sesuatu pesan di dalam media. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, audio, teks, dan video [4].

2. METODE PENELITIAN

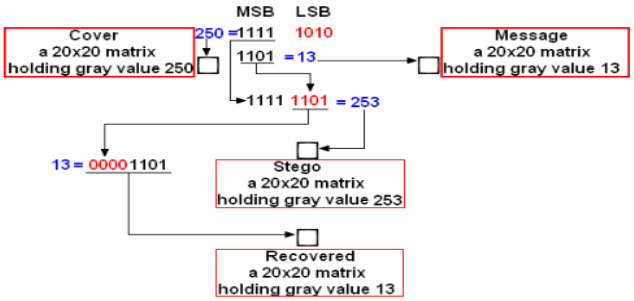
Teknik kriptografi ini diciptakan dengan maksud untuk dapat menciptakan cipher (kode) yang tidak dapat dipecahkan menggunakan teknik analisis frekuensi. Hill Cipher tidak mengganti setiap abjad yang sama pada plaintext dengan abjad lainnya yang sama pada ciphertext karena menggunakan perkalian matriks pada dasar enkripsi dan dekripsinya. Oleh karena itu Hill Cipher termasuk dalam salah satu kriptosistem polialfabetik. Cipher ini ditemukan pada tahun 1929 oleh Lester S. Hill. Teknik enkripsi yang digunakan adalah enkripsi simetris kunci dekripsi sama dengan kunci enkripsi. Untuk *public key cryptography,* diperlukan teknik enkripsi asimetris kunci dekripsi tidak sama dengan kunci enkripsi [5]. Enkripsi, dekripsi dan pembuatan kunci untuk teknik enkripsi asimetris memerlukan komputasi yang lebih intensif dibandingkan enkripsi simetris, karena enkripsi asimetris menggunakan bilangan-bilangan yang sangat besar. Walaupun enkripsi asimetris lebih “mahal" dibandingkan enkripsi simetris, *public key cryptography* sangat berguna untuk *key management* dan *digital signature.* suatu proses enkripsi yang baik menghasilkan naskah acak yang memerlukan waktu yang lama (contohnya satu juta tahun) untuk didekripsi oleh seseorang yang tidak mempunyai kunci dekripsi [6]. Semakin banyak proses yang diperlukan berarti juga semakin lama waktu yang dibutuhkan, maka semakin kuat algoritma tesebut dan semakin aman digunakan untuk menyandikan pesan [7]. Dasar dari teknik Hill Cipher adalah aritmatika modulo terhadap matriks. Dalam penerapannya, Hill Cipher menggunakan teknik perkalian matriks dan teknik invers terhadap matriks. Kunci pada Hill Cipher adalah matriks n x n dengan n merupakan ukuran blok [8]. Matriks K yang menjadi kunci ini harus merupakan matriks yang invertible, yaitu memiliki inverse K-1 sehingga kunci harus memiliki invers karena matriks K- adalah kunci yang digunakan untuk melakukan dekripsi [9]. Proses enkripsi pada Hill Cipher dilakukan per blok plaintext. Ukuran blok tersebut sama dengan ukuran matriks kunci. Sebelum membagi teks menjadi deretan blok-blok, plaintext terlebih dahulu dikonversi menjadi angka, masing-masing sehingga A=0, B=1, hingga Z=25. Secara matematis, proses enkripsi pada Hill Cipher adalah: C = K . P; dengan C = Ciphertext K = Kunci P = Plaintext.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
13	14	15	16	17	18	19	20	21	22	23	24	25

Proses dekripsi pada Hill Cipher:
C = K . P
K^{-1} . C = K^{-1} . K . P
K^{-1} . C =I . P
P = K^{-1} . C

Menjadi persamaan proses deskripsi: P= K^{-1} . C untuk menentukan K^{-1}: ; $\frac{1}{det\,det\,K} mod\ 26 = x$ atau (det K * x mod 26 = 1).

Least Significant Bit adalah salah satu metode untuk menyembunyikan pesan dalam media digital dengan cara menyisipkan pesan tersebut pada satu bit paling kanan ke pixel file obyek. Dalam menyisipkan data pesan ke dalam berkas citra digital dengan menggunakan metode *Least Significan Bit* (LSB) *Modification* [10]. Modul proses output data akan memisah kembali antara file citra digital dan data pesan rahasia pada suatu stegano image serta melakukan dekripsi pesan menjadi *ciphertext* [11].

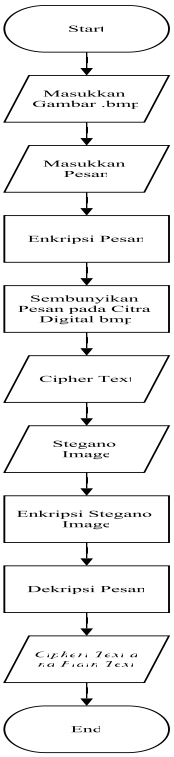


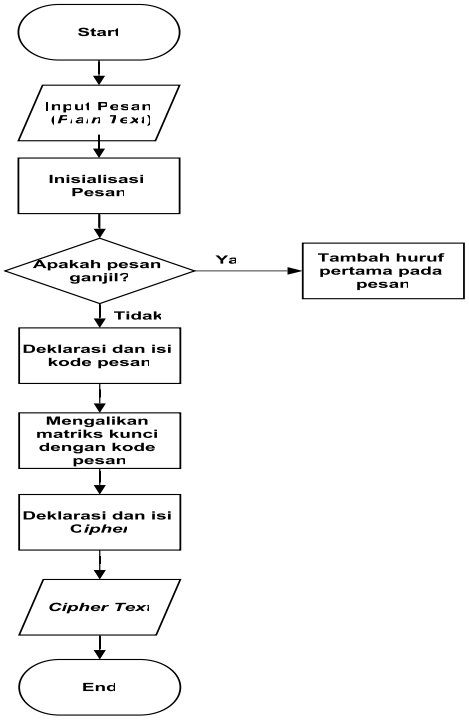
Gambar 1. Mekanisme LSB

Gambar 1 menunjukkan penerapan LSB menggunakan media gambar berbasis pixel dengan nilai 8 bit (*gray value*). Setiap pixel yang terdiri dari 8 bit dibagi menjadi 2 bagian yaitu, 4 bit MSB (*most significant bit*) dan 4 bit LSB (*least significat bit*).

3. HASIL DAN PEMBAHASAN

Dalam perancangan algoritma digunakan pendekatan terstruktur (structured approach) (Gambar 2). Algoritma Hill Cipher digunakan untuk menentukan posisi enkrip plaintext menjadi ciphertext yang secara acak (Gambar 3).





Gambar 2. Algoritma Umum Perangkat Lunak

Gambar 3. Algoritma Hill Cipher

Proses enkripsi pada hill cipher dilakukan per blok plaintext. Ukuran blok tersebut sama dengan ukuran matriks kunci. Sebelum membagi teks menjadi deretan blok-blok, plaintext dikonversi menjadi angka, masing-masing sehingga A=1, B=2, hingga y-25. Z diberi nilai 0.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
13	14	15	16	17	18	19	20	21	22	23	24	25

Proses enkripsi pada *hill cipher* :
C = K.P(2);
C = Ciphertext
K = Kunci
P = Plaintext
Jika terdapat *plaintext* P:
P = SELAMAT
Maka *plaintext* tersebut dikonversi menjadi:
P = 18 4 11 0 12 0 19
Plaintext tersebt akan dienkripsi dnegan teknik *hill cipher*, dengan kunci K yang merupakan matriks 2x2.

$$K = \begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix}$$

Matrik kunci K berukuran 2x2, maka *plaintext* dibagi menjadi blok yang masing-masing bloknya berukuran 2 karakter. Karena karakter terakhir tidak ada memiliki pasangan karakter yang sama yaitu W.P menjadi SELAMAT. Blok pertama dari *plaintext* P adalah:

$$P_{1,2} = \begin{bmatrix} 18 \\ 4 \end{bmatrix}$$

Blok *plaintext* dienkripsi dengan kunci K :

$$C_{1,2} = \begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} \begin{bmatrix} 18 \\ 4 \end{bmatrix} = \begin{bmatrix} 38 \\ 194 \end{bmatrix}$$

Hasil perhitungan menghasilkan angka yang tidak berkorespondensi dengan huruf, maka lakukan modulo 26 pada hasil sehingga $C_{1,2}$ menjadi:

$$C_{1,2} = \begin{bmatrix} 38 \\ 194 \end{bmatrix} = \begin{bmatrix} 12 \\ 12 \end{bmatrix} (mod\ 26)$$

Karakter yang berkorespondensi dengan 12 dan 12 adalah M dan M, maka S menjadi M dan E menjadi M. Setelah melakukan enkripsi semua blok pada *plaintext* P maka menghasilkan *ciphertext* C : P = SELAMAT ; C = 12 12 11 21 12 4 10 11 ; C = MMLVMEKL
Dari *ciphertext* yang dihasilkan terlihat bahwa *hill cipher* menghasilkan *ciphertext* yang tidak memiliki pola yang mirip dengan *plaintext*-nya. Proses dekripsi pada *hill cipher* pada dasarnya sama dengan proses enkripsi. Namun matriks kunci harus dibalik (invers) terlebih dahulu. Secara matematis, proses dekripsi pada *hill cipher* :

$$C = K.P\ ;\ K^{-1}.C = K^{-1}.K.P\ ;\ K^{-1}.C = I.P\ ;\ P = K^{-1}.C$$

Proses dekripsi : $P = K^{-1}.C$; menggunkan kunci $K = \begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix}$
Maka, proses dekripsi diawali dnegan mencari invers dari matriks *K*. Mencari invers dapat dilakukan dengan menggunakan metode operasi baris (*row operation*) atau metode determinan. Setelah melakukan perhitungan, didapat matriks K^{-1} yang merupakan invers dari matriks *K*, yaitu:

$$K^{-1} = \begin{bmatrix} 27 & -54 \\ -18 & 45 \end{bmatrix} = \begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix} (mod\ 26)$$

Kunci K^{-1} yang digunakan untuk melakukan dekripsi ini telah memnuhi persamaan (1) karena:

$$K.\ K^{-1} = \begin{bmatrix} 53 & 234 \\ 26 & 105 \end{bmatrix} = K^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} (mod\ 26) = I$$

Ciphertext C = MMLVMEKL, akan didekripsi dengan menggunakan kunci dekripsi K^{-1} dengan persamaan (3). Proses dekripsi ini dilakukan blok per blok seperti pada proses enkripsi. Pertama-tama ubah huruf-huruf pada *ciphertext* urutan numerik. C = 12 12 11 21 12 4 10 11
Proses dekripsi dilakukan sebagai berikut:

$$P_{1,2} = K^{-1}.C_{1,2}\ ;\ P_{1,2} = \begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix} \begin{bmatrix} 12 \\ 12 \end{bmatrix} = \begin{bmatrix} 252 \\ 264 \end{bmatrix} (mod\ 26) = \begin{bmatrix} 18 \\ 4 \end{bmatrix}$$

$P_{3,24} = K^{-1}.C_{3,4}$; $P_{3,24} = \begin{bmatrix}4 & 17\\15 & 7\end{bmatrix}\begin{bmatrix}11\\21\end{bmatrix} = \begin{bmatrix}401\\312\end{bmatrix}(mod\ 26) = \begin{bmatrix}11\\0\end{bmatrix}$

Setelah semua blok selesai didekripsi, maka didaapkan hasil *plaintext*: P = 18 4 11 0 12 0 19;
P = SELAMAT

Least Significant Bit adalah salah satu metode untuk menyembunyikan pesan dalam media digital dengan cara menyisipkan pesan tersebut pada satu bit paling kanan ke pixel file obyek. Dalam menyisipkan data pesan ke dalam berkas citra digital dengan menggunakan metode *Least Significan Bit* (LSB) *Modification*. Misalkan untuk menyisipkan suatu segmen pesan hasil dan modulasi sebesar 4 byte dengan modifikasi 1 bit LSB, maka dibutuhkan 32 data citra digital untuk menampungnya. Dari segmen pesan ' 1 0 1 0 ' dengan 4 byte data citra digital: 0 1101110 00100011 01000010 01101101'.

Dengan operasi penggantian bit terakhir dengan 4 bit segmen pesan secara berurutan menjadi:
Data citra digital: '0 1 1 0 1 1 1 0 0 0 1 0 0 0 1 1 0 10 0 0 0 1 0 0 1 1 0 1 1 0 1'
Pesan: 1 0 1 0
Hasil: '0 1 1 0 1 1 1 <u>1</u> 0 0 1 0 0 0 1 <u>0</u> 0 1 0 0 0 0 1 <u>1</u> 0 1 1 0 1 1 0 <u>0</u>'

Dengan sedikit modifikasi ini, maka efek dari perubahan nilai warna yang terjadi akibat perubahan bit tersebut tidak terlalu berpengaruh terhadap kualitas gambar. Perhatikan contoh untuk menyisipkan sebuah karakter A ke dalam citra *grayscale*. Sebuah pesan huruf A akan disisipkan ke dalam citra *grayscale* 8 bit ukuran 10x10 piksel.

1	6	5	3	7	4	7	4	1	0
3	5	3	5	5	5	5	7	7	0
0	0	0	2	2	6	6	6	6	6
5	5	4	4	4	4	4	4	7	3
2	2	0	0	0	0	1	1	1	1
7	5	5	5	7	7	7	6	3	3
3	3	3	3	3	3	3	3	7	5
5	5	5	5	5	5	5	5	2	3
0	0	0	0	0	0	4	4	4	4
3	3	3	3	3	1	1	1	6	2

Langkah pertama adalah mengunbah kedua data tersebut (huruf A dan citra) menjadi biner. Nilai biner untuk A adalah 10000011. Karena jumlah digit biner huruf A hanya 8 bit maka jumlah piksel citra *grayscale* yang dibutuhkan cukup 8 piksel saja. 8 piksel pertama dari citra yang diubah menjadi biner.

8 piksel pertama

1	6	5	3	7	4	7	4	1	0
3	5	3	5	5	5	5	7	7	0
0	0	0	2	2	6	6	6	6	6
5	5	4	4	4	4	4	4	7	3
2	2	0	0	0	0	1	1	1	1
7	5	5	5	7	7	7	6	3	3
3	3	3	3	3	3	3	3	7	5
5	5	5	5	5	5	5	5	2	3
0	0	0	0	0	0	4	4	4	4
3	3	3	3	3	1	1	1	6	2

Langkah kedua mengganti bit terakhir dari piksel citra dengan bit dari huruf A 1 piksel media.

Tabel 1 piksel citra yang diambil

Piksel Citra		Huruf A
Decimal	Biner	
1	00000001	1
6	00000110	0
5	00000101	0
3	00000011	0
7	00000111	0
4	00000100	0
7	00000111	1
4	00000100	1

piksel citra yang berubah	
Decimal	Biner
1	00000001
6	00000110
4	00000110
2	00000100
6	00000010
4	00000110
7	00000111
5	00000101

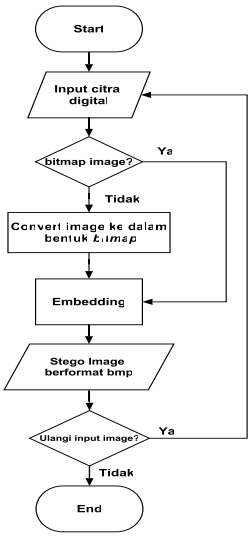
bit-bit yang ditandai dengan kotak. Bit-bit piksel citra mengalami perubahan;

8 Bit yang pertama

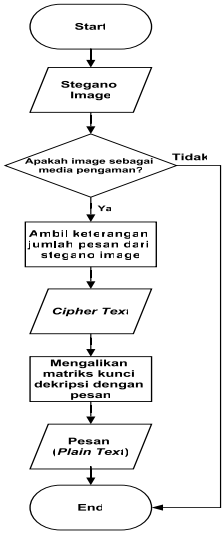
1	6	4	2	6	4	7	5	1	0
3	5	3	5	5	5	5	7	7	0
0	0	0	2	2	6	6	6	6	6
5	5	4	4	4	4	4	4	7	3
2	2	0	0	0	0	1	1	1	1
7	5	5	5	7	7	7	6	3	3
3	3	3	3	3	3	3	3	7	5
5	5	5	5	5	5	5	5	2	3
0	0	0	0	0	0	4	4	4	4
3	3	3	3	3	1	1	1	6	2

penampung = 8 bit. Untuk menampung 1 bit data pesan diperlukan 1 piksel citra media penampung berukuran 8 bit karena setiap 8 bit hanya bisa menyembunyikan satu bit di LSB-nya. Oleh karena itu, citra ini hanya mampu menampung data pesan sebesar maksimum 16384/8 = 2048 bit dikurangi panjang nama filenya karena penyembunyian data rahasia tidak hanya menyembunyikan isi data tersebut, tetapi juga nama filenya. Semakin besar data yang disembunyikan di dalam citra, semakin besar pula kemungkinan data tersebut rusak akibat manipulasi pada citra penampung, media penyimpanan hasil steganografi adalah citra digital berformat bitmap. Proses ini merupakan memanipulasi pesan chiperteks heksadesimal hasil enkripsi yang disisipkan dalam bitmap. Citra digital yang diinput dalam berbagai format yang nantinya pada proses steganografi akan dikonversikan dalam bentuk bitmap, Gambar 4. Proses Penyisipan Pesan, cover image yang digunakan format bitmap. Untuk lebih Jelasnya proses penyisipan pesan dan Penyisipan pesan steganografi LSB pada citra digital, Gambar 5. Proses extraction dilakukan hampir sama halnya dengan proses embedding atau penyisipan tetapi ada beberapa perbedaan yakni proses ini terlebih dahulu dipilih stego object yaitu image yang sudah disisipi sebuah pesan dan mendekripsi chipertext menjadi plaintext, Gambar 6.

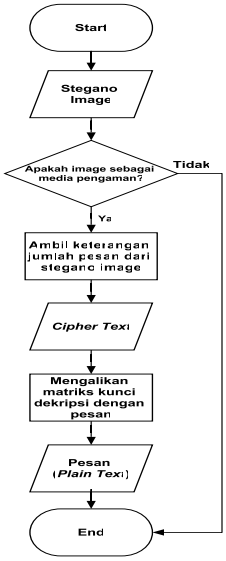
.



Gambar 4. Proses Pemilihan Citra Digital

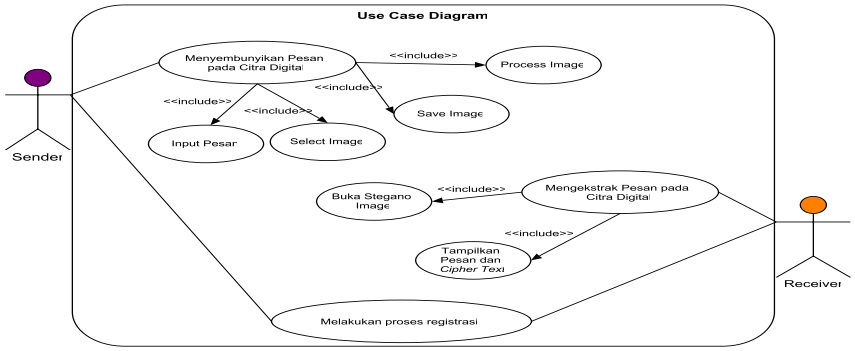


Gambar 5. Penyisipan Pesan



Gambar 6. Proses Ekstrasi Pesan

Use case diagram (Gambar 7), *user* pengirim memilih citra yang akan disisipi oleh teks. Kemudian *user* pengirim menuliskan pesan yang akan disembunyikan di dalam citra tersebut. Hasilnya adalah berupa citra yang telah disisipi oleh teks (*stego image).* Sedangkan *user* penerima adalah menerima *stego image,* kemudian memasukkan *password,* dan melakukan ekstraksi, sehingga pesan yang disisipkan dan disembunyikan akan muncul.



Gambar 7. Use Case Diagram

Implementasi sistem merupakan tahapan dalam menerapkan sistem yang telah dibangun, dimana nantinya akan diketahui kualitas dari sistem yang dirancang. Dalam melakukan implementasi dipersiapkan beberapa sarana yang berhubungan dengan perangkat keras (*hardware*) dan perangkat lunak (*software*). Aplikasi dirancang untuk membantu menyembunyikan pesan teks yang bersifat rahasia pada sebuah citra digital dengan format *bitmap*. Tampilan antar muka (*interface*) dari aplikasi pengamanan pesan pada citra digital dengan menggabungkan teknik kriptografi dan steganografi. *Form* Penyisipan Pesan (*Encode*) merupakan tampilan dimana *user* mamasukan/*input* citra digital dan pesan teks untuk proses pengamanan pesan teks kedalam suatu citra digital, Gambar 8. *Form Decode* merupakan tampilan dimana user mengembalikan data teks yang telah di enkripsi kebentuk pesan semula, Gambar 9.

Gambar 8. *Form Encode*

Gambar 9. *Form Decode*

Pengujian Algoritma *Hill Cipher*

Sebelum masuk kedalam proses penyandian terlebih dahulu ditetapkan pesan yang akan disandikan dan kunci matriks 2x2. Pesan yang disandikan maksimal 66 karakter tiap karakter harus berada diantara A-Z yang berjumlah 26 huruf dalam proses penyandian ini huruf besar dan kecil tidak dibedakan. Dan juga dalam penyandian ini tidak menggunakan kode ascii huruf tetapi dengan kode angka berdasarkan urutan huruf yaitu A=0, B=1, ... Z=25 dan spasi tidak dihitung dalam penyandian. Kunci yang telah ditetapkan digunakan di dalam proses enkripsi pesan berikut ini adalah langkah-langkah penyandian pesan mengunakan kunci matriks 2x2 :

1. Sisipkan pesan dan kunci matrik

   Pesan : P = SELAMAT Kunci : K = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix}$

Untuk mengetahui kode-kode dari pesan akan diberikan tabel kode masing-masing huruf dari A sampai dengan Z.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
13	14	15	16	17	18	19	20	21	22	23	24	25

2. Mengubah pesan menjadi kode dan matrik 2x2. Setelah diketahui masing-masing kode huruf maka pesan teks yang akan di sandikan diubah kedalam kode-kode angka dari tabel diatas, yaitu : Kode pesan : P = 18 4 11 0 12 0 19 19

Kemudian setelah didapat kode untuk masing-masing huruf kemudian setiap dua kode diubah kedalam matriks ordo 2 x 1, agar dapat dikalikan dengan kunci yang mempunyai matriks 2x2.

$$P = \begin{bmatrix} 18 \\ 4 \end{bmatrix} \begin{bmatrix} 11 \\ 0 \end{bmatrix} \begin{bmatrix} 12 \\ 0 \end{bmatrix} \begin{bmatrix} 19 \\ 19 \end{bmatrix}$$

Setelah matrik disusun maka Rumus: C=(P*K) mod 26

3. Mengalikan matriks pesan dan matriks kunci

Matriks pesan dikalikan dengan matriks kunci, dan hasil perkalikan tersebut diubah lagi kedalam huruf dengan referensi tabel kode masing-masing huruf. Proses perkalian matriks kunci dengan matriks pesan :

C = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} x \begin{bmatrix} 18 \\ 4 \end{bmatrix} = \begin{bmatrix} 38 \\ 194 \end{bmatrix}$

Karena hasil perkalian melebihi 25 maka hasil perkalian ini harus di mod 26 :

C = $\begin{bmatrix} 38 \\ 194 \end{bmatrix} Mod\ 26 = \begin{bmatrix} 12 \\ 12 \end{bmatrix}$; untuk K*P (LA)

C = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} x \begin{bmatrix} 11 \\ 0 \end{bmatrix} = \begin{bmatrix} 11 \\ 99 \end{bmatrix} mod\ 26 = \begin{bmatrix} 11 \\ 21 \end{bmatrix}$; untuk K*P (MA)

C = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} x \begin{bmatrix} 12 \\ 0 \end{bmatrix} = \begin{bmatrix} 12 \\ 108 \end{bmatrix} mod\ 26 = \begin{bmatrix} 12 \\ 4 \end{bmatrix}$; untuk K*P (TT)

C = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} x \begin{bmatrix} 19 \\ 19 \end{bmatrix} = \begin{bmatrix} 114 \\ 323 \end{bmatrix} mod\ 26 = \begin{bmatrix} 10 \\ 11 \end{bmatrix}$

Setelah semua matriks pesan dikalikan dengan matriks kunci dan lakukan modulus dengan 26 hasil matrik *cipher*

$\begin{bmatrix} 12 \\ 12 \end{bmatrix} \begin{bmatrix} 11 \\ 21 \end{bmatrix} \begin{bmatrix} 12 \\ 4 \end{bmatrix} \begin{bmatrix} 10 \\ 11 \end{bmatrix}$

4. Mengubah matriks menjadi deret pesan. Setelah didapat hasil matriks dari perkalian antara matriks kunci dan pesan kemudian matriks disusun kembali berurutan:

C = 12, 12, 11, 21, 12, 4, 10, 11

5. Mengubah kode pesan menjadi huruf (karakter)

Kode diatas disusun kembali kedalam bentuk huruf dengan menggunakan tabel 5.1. Dan hasil ini adalah chiper hasil dari penyandian pesan yang akan di gunakan dalam proses steganografi, yaitu Ciphertext: MMLVMEKL

Setelah proses enkripsi selesai dilakukan, maka untuk mendeskripsi cipherteks menjadi pesan kembali sebenarnya hampir sama dengan cara enkripsi. Tetapi kunci yang digunakan harus di invers terlebih dahulu. Untuk lebih jelasnya tentang proses dekskripsi chiperteks akan dijelaskan secara rinci tentang tahap-tahap deksripsi.

1. Invers Matriks Kunci

Pengembalian pesan terdapat ketetapan yang telah ditentukan tahap pertama yaitu dengan melakukan invers terhadap matriks kunci yang digunakan dalam penyandian pesan. Rumus untuk menginvers matriks yang ber ordo 2x2. Proses untuk mengetahui invers matriks kunci invers

$$K^{-1} = \begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix}$$

Setelah invers matriks kunci selesai dilakukan maka didapat matriks. Kunci-1 untuk mendeskripsikan chiper menjadi pesan kembali matrik invers kunci dikalikan dengan matriks chipperteks. Ubah chipperteks ke dalam bentuk kode kembali sama persis seperti proses pengubahan pesan kedalam kode sewaktu proses penyandian.

2. Menyiapkan Pesan Cipher

Ciphertext: MMLVMEKL

Setelah diketahui cipherteks, ubah kedalam bentuk kode berdasarkan urutan angka.

3. Mengubah Chiper Menjadi Kode Dan Matriks

C = 12, 12, 11, 21, 12, 4, 10, 11

Setelah diketahui masing-masing kode dari cipherteks, kemudian diubah kedalam bentuk matriks *cipher*:

C =$\begin{bmatrix} 12 \\ 12 \end{bmatrix} \begin{bmatrix} 11 \\ 21 \end{bmatrix} \begin{bmatrix} 12 \\ 4 \end{bmatrix} \begin{bmatrix} 10 \\ 11 \end{bmatrix}$

Setelah diketahui matriks chipper kemudian chiper dikalikan dengan invers matrik kunci dan modulus dengan 26 dan hasil itu adalah pesan asli dari penyandian yang telah dilakukan sebelumnya. Rumus untuk menentukan pesan teks dari chiperteks P = K-1*C.

4. Mengalikan matriks pesan chipper dengan invers matriks kunci. Proses perkalian antara invers matriks kunci dengan matriks chipperteks, untuk kata MM:

P =$\begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix} x \begin{bmatrix} 12 \\ 12 \end{bmatrix} = \begin{bmatrix} 252 \\ 264 \end{bmatrix} mod\ 26 = \begin{bmatrix} 18 \\ 4 \end{bmatrix}$; untuk kata LV:

P =$\begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix} x \begin{bmatrix} 11 \\ 21 \end{bmatrix} = \begin{bmatrix} 401 \\ 312 \end{bmatrix} mod\ 26 = \begin{bmatrix} 11 \\ 0 \end{bmatrix}$; untuk kata ME:

P = $\begin{bmatrix}4 & 17\\15 & 7\end{bmatrix} x \begin{bmatrix}12\\4\end{bmatrix} = \begin{bmatrix}116\\208\end{bmatrix} mod\ 26 = \begin{bmatrix}12\\0\end{bmatrix}$; untuk kata KL:

P = $\begin{bmatrix}4 & 17\\15 & 7\end{bmatrix} x \begin{bmatrix}10\\11\end{bmatrix} = \begin{bmatrix}227\\227\end{bmatrix} mod\ 26 = \begin{bmatrix}19\\19\end{bmatrix}$

Dan seletelah semua matriks chipper kalikan dengan inverst matriks kunci K-1 maka hasil perkalian dan modulus 26:

$$P = \begin{bmatrix}18\\4\end{bmatrix}\begin{bmatrix}11\\0\end{bmatrix}\begin{bmatrix}12\\0\end{bmatrix}\begin{bmatrix}19\\19\end{bmatrix}$$

5. Mengubah matriks pesan menjadi deret kode

Setelah diketahui masing-masing matriks pesan kemudian matriks pesan diurutkan kedalam bentuk bilangan bulat biasa P = 18, 4, 11, 0, 12, 0, 19, 19

6. Mengubah kode menjadi pesan kembali

Langkah terakhir adalah mengubah pesan yang masih berbentuk kode menjadi bentuk huruf dengan berpedoman pada tabel kode masing-masing huruf yang tertera pada tabel 5.1, hasilnya:

P = HALO; Pengujian Metode *Least Significant Bit* (LSB)

Proses penyisipan pesan chiper kedalam citra gambar. Gambar yang digunakan adalah gambar berwarna 24 bit, yaitu gambar yang terdiri dari 3 warna R, G, B masing-masing warna mempunyai kedalaman warna sebesar 8 bit. Karena masing-masing warna bernilai 8 bit, maka pesan akan disisipkan kedalam bit R, bit G dan bit B tiap-tiap pixel. Misalkan pesan yang akan disisipkan sebanyak 8 bit, maka pesan yang 8 bit tersebut hanya akan disisipkan pada dua 3 pixel, karena tiap pixel memiliki kapasitas 24 bit dan masing-masing bit pesan hanya disisipkan pada 8 bit citra gambar. Dibawah ini adalah langkah-langkah proses steganografi untuk menyisipkan pesan kedalam citra gambar, Berikut ini merupakan bagaimana cara kerja dari algoritma LSB dimana teks HALO akan disisipkan kedalam gambar, namun terlebih dahulu teks tersebut diubah kedalam biner dengan nilai. Tabel 2.

Tabel 2. Kode ASCII Pesan yang akan disisipi

Teks	Biner
S	00010010
E	00000100
L	00001011
A	00000000
M	00001100
A	00000000
T	00010011

Setelah diubah kedalam biner lalu pesan akan disisipkan kedalam gambar pada warna (RGB) dengan metode LSB dengan nilai biner gambar awal Tabel 3.

Tabel 3. Kode media/gambar yang akan disisip

01110111	01110110	01110100	01000111
01110011	01110100	01110110	01110000
00110110	01110111	11110111	01110110
10110111	11110111	01110111	11110111
11010111	01110110	11110111	01110110
11110111	10110111	01111111	01110111
01110100	11000111	11110111	00010111
01111111	11010111	01111111	01110100

disisipkan pesan maka nilai biner gambar akan berubah menjadi Tabel 4.

Tabel 4. Kode media/gambar yang sudah disisip

0111011**0**	01110110	01110100	0100011**0**
01110011	0111010**1**	0111011**1**	0111000**1**
00110110	01110111	11110111	0111011**1**
1011011**0**	1111011**0**	0111011**0**	1111011**0**
11010111	01110110	11110111	0111011**1**
1111011**0**	1011011**0**	01111111	01110111
01110100	1100011**0**	1111011**0**	00010111
0111111**1**	11010111	0111111**0**	0111010**1**

Proses ekstraknya adalah mengambil nilai paling kanan dari biner yang disisipkan. Data biner yang telah diambil isi pesannya Tabel 5.

Tabel 5. Kode ASCII hasil ekstrak

Teks	Biner
S	00010010
E	00000100
L	00001011
A	00000000
M	00001100
A	00000000
T	00010011

Beberapa pengujian yaitu pengujian hasil teori dan hasil praktek (pengaplikasian), lalu pengujian perbandingan gambar sebelum dan sesudah pesan disisipkan.

Tabel 6. Analisa Hasil Pada Enkripsi Hill Cipher

ANALISA HASIL PADA ENKRIPSI *HILL CIPHER*		
Hill Cipher	Teori	Praktek
Pesan	SELAMAT	MMLVMEKL
Hasil	Huruf S menjadi M Huruf E menjadi M Huruf L menjadi L Huruf A menjadi V Huruf M menjadi M Huruf A menjadi E Huruf T menjadi K Sehingga kata SELAMAT menjadi kata MMLVMEKL	Dalam prakteknya dengan menekan tompol submit maka hasilnya adalah sebagai berikut: 

Pengujian selanjutnya adalah analisa hasil sebelum dan sesudah enkripsi/disisipkan pesan pada citra BMP. Adapun pesan yang disisipkan adalah “Proses mengubah citra analog menjadi citra digital disebut digitalisasi citra. Ada dua hal yang harus dilakukan pada digitalisasi citra, yaitu digitalisasi spasial yang disebut juga sebagai sampling (penerokan) dan digitalisasi intensitas yang sering disebut sebagai kuantisasi” Tabel 7.

Tabel 7. Perbandingan Citra BMP

ANALISA HASIL PADA CITRA BMP		
Detail	Citra Asli	Citra Stegano
Nama Citra	logo aplikasi.bmp	CITRA1.bmp
Ukuran (Mb)	1.20	1.20
Dimensi (Pixel)	650 x 650	650 x 650

Citra		

IV. KESIMPULAN

Berdasakan dari analisa, perancangan dan implementasi pada aplikasi pengamanan pesan pada citra digital dengan menggabungkan metode *Least Significant Bit* (LSB) dan algoritma *hill cipher*, dapat diambil kesimpulan:

1. Proses pengamanan pesan pada citra digital aman dan tidak diketahui secara kasat mata, karena besar dari *bitmap* hasil steganografi tidak terlihat secara signifikan perubahan setelah dari proses penyisipan biner teks ke dalam biner *bitmap* menggunakan metode *least significant bit* (LSB) yaitu penggantian bit terakhir sehingga kapasitas dari *bitmap* sebelum dan sesudah disteganografi tidak mengalami perubahan yang signifikan.
2. Pengujian pesan teks menggunakan algoritma *hill cipher* berhasil dilakukan sesuai tepat dengan alur atau langkah-langkahnya sehingga menghasilkan cipherteks yang berupa pengacakan huruf abjad.
3. Pesan yang akan diambil dari bitmap dapat dilanjutkan ke proses dekripsi yang bertujuan untuk mengembalikan pesan *ciphet text* ke bentuk semula (*plain text*) melalui proses dekripsi algoritma *hill cipher* yang sesuai.

DAFTAR PUSTAKA

M. M. Amin, 2016. Implementasi Kriptografi Klasik Pada Komunikasi Berbasis Teks, Jurnal Pseudocode, Volume III Nomor 2, September hal. 129-136.

Supiyanto. 2015. Implementasi Hill Cipher Pada Citra Menggunakan Koefisien Binomial Sebagai Matriks Kunci, Supiyanto, Seminar Nasional Informatika UPN ”Veteran” Yogyakarta, hal. 284-292.

Anita S, RMS. 2019. Sistem Bilangan Digital, 1, Serang Banten, CV. AA. Rizky.

Satriya, T, C, K., Dedih, Supriyadi, Implementasi Kriptografi Algoritma Rivest Shamir Adleman dengan Playfair Cipher pada Pesan Teks Berbasis Android, JOIN (Jurnal Online Informatika), Volume 2 No. 2, hal : 102-109.

Abdul H, H. 2013. Implementasi Algoritma Hill Cipher Dalam Penyandian Data, Pelita Informatika Budi Darma, Volume : IV, Nomor : 2.

Anita, S, RM, Sinaga. 2017. Implementasi Teknik Threshoding Pada Segmentasi Citra Digital, Jurnal Manajemen Dan Informatika Pelita Nusantara, Volume 1 No 2 hal : 48-51.

Indra, G. Sumarno. Eka, I. Heru, S, T. 2017. Pengamanan Berkas Dokumen Menggunakan Fungsi Algoritma Steganografi LSB, Jurnal Informatika Mulawarman, Vol. 12, No. 2.

Niria, L. Anita, S, RMS. 2018. Implementasi Steganografi LSB Dengan Enkripsi Vigenere Cipher Pada Citra, ScientiCO : Computer Science Informatics Journal Vol. 1, No. 2, hal : 47-58.

Anita, S, S. 2017. SEGMENTASI RUANG WARNA L*a*b Jurnal Mantik Penusa Vol. 3, No. 1 , pp. 43-46.

Rohmat, N, I. Ilham. MS. 2017. PERANCANGAN APLIKASI STEGAKRIP DENGAN METODE LSB DAN ALGORITMA RSA BERBASIS WEB, Jurnal Computech & Bisnis, Vol. 11, No 1, 98-109.

JISKa, Vol. 4, No. 3, JANUARI, 2020, Pp. 149 – 155
ISSN : 2527 – 5836 (print) | 2528 – 0074 (online)

Pengembangan Aplikasi *Computer Based Test* dengan Protokol *Two Central Facilities*

Muhammad Nasyithul Ibad [(1)], Syarif Alqoroni [(2)],
Muhammad Ammarullah Ridho [(3)] Khadijah Fahmi Hayati Holle[(4)]
Jurusan Teknik Informatika – UIN Maulana Malik Ibrahim Malang
JL. Gajayana No. 50, Dinoyo, Kec. Lowokwaru, Kota Malang - Indonesia
e-mail : muhammad.nasyithul.ibad@gmail.com[(1)], syarifalqoroni2@gmail.com[(2)],
study.ammar@gmail.com[(3)] khadijah.holle@uin-malang.ac.id[(4)]

Abstract

The Indonesian government has issued a policy on Ujian Nasional Berbasis Komputer (UNBK), or it can be called CBT (Computer Based Test). In this study is the development of the CBT system using the Two Central Facilities protocol, which consists of the Central Legitimization Agency (CLA) and the Central Tabulating Facilities (CTF). In developing this CBT system, CLA is used to authenticate the examinees, while CTF is used to provide questions and calculation of exam answers. To maintain security added encryption using the RSA Algorithm (Rivest-Shamir-Adleman) which functions to convert data into ciphertext. From the results of functional trials, this system has succeeded in applying the Two Central Facilities protocol. This system has been connected to the CLA and CTF servers and all data has been successfully encrypted. For this reason, in the future schools will be endeavored to implement the development of the CBT system to avoid fraud.

Keywords : UNBK, Two Central Facilities, RSA Encryption

Abstrak

Ujian Nasional merupakan suatu ujian yang menjadi tolak ukur pemahaman dari seorang siswa, akan tetapi Ujian Nasional secara tertulis rawan terjadinya berbagai kecurangan. Oleh karena itu pemerintah mengeluarkan kebijakan tentang Ujian Nasional Berbasis Komputer (UNBK) atau bisa disebut CBT (Computer Based Test). Dalam penelitian ini merupakan pengembangan sistem CBT dengan menggunakan protokol *Two Central Facilities*, yang terdiri dari *Central Legitimization Agency* (CLA) dan *Central Tabulating Facilities* (CTF). Dalam pengembangan sistem CBT ini, CLA digunakan untuk autentikasi peserta ujian, sedangkan CTF digunakan untuk penyedia soal dan perhitungan jawaban ujian. Untuk menjaga keamanan ditambahkan enkripsi menggunakan Algoritma RSA (*Rivest-Shamir-Adleman*) yang berfungsi untuk merubah data menjadi cipherteks. Dari hasil uji coba fungsional, pada sistem ini telah berhasil dalam menerapkan protokol Two Central Facilities. Sistem ini telah terhubung dengan server CLA dan CTF serta semua data telah berhasil terenkripsi menggunakan Algoritma RSA, sehingga semua terjamin keamananya. Untuk itu, kedepannya sekolah-sekolah diusahakan menerapkan hasil pengembangan sistem CBT ini.

Kata Kunci : UNBK, Two Central Facilities, Enkripsi RSA

1. PENDAHULUAN

Berkembangnya teknologi menjadikan banyak proses yang beralih ke media digital dan komputerisasi, banyak pekerjaan yang awalnya dikerjakan oleh manusia mulai tergantikan oleh mesin dan komputer. Fenomena ini terjadi di berbagai sektor kehidupan. Salah satu perubahan yang terjadi yaitu di bidang pendidikan. Perubahan yang kita ketahui yakni bergantinya Ujian Nasional (UN) berbasis ujian tulis menjadi Ujian Nasional Berbasis Komputer (UNBK) atau lebih umum dikenal dengan istilah CBT (*Computer Based Test*).

UNBK pertama kali diselenggarakan pada tahun 2014 secara online dan dibatasi di SMP Indonesia Singapura dan SMP Indonesia Kuala Lumpur (SIKL). Hasil dari penyelenggaraan UNBK pada kedua sekolah tersebut menghasilkan nilai yang memuaskan, sehingga mendorong untuk meningkatkan pengetahuan siswa terhadap Teknologi Informasi dan Komunikasi (TIK). (Kemendikbud, 2019)

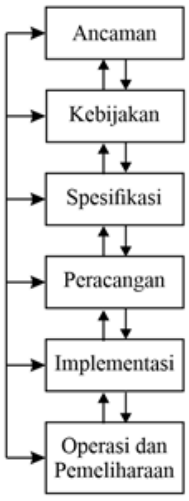
Laporan dan hasil CBT lebih cepat diolah, namun berbagai kecurangan tidak akan terhindarkan pada sistem UNBK (Poggio, Glassnapp, & Yang, 2005). Oleh karena itu sistem UNBK yang dibuat harus memenuhi standar untuk menjamin keamanan pada setiap ancaman yang akan terjadi. Menurut pendapat Bruce Schneir (1996), salah satu solusi dalam menjaga keamanan data adalah dengan adanya protokol *Two Central Facilities* yang terdiri dari *Central Legitimization Agen* (CLA) untuk autentikasi user dan *Central Tabulating Facilities* untuk perhitungan data Schneir, 1996). Siressha dan Chackai (2005) pada bukunya *Secure Virtual Election Booth with Two Central Facilities*, juga memaparkan desain protokol *Secure Election* dengan *Two Central Facilities*, yaitu *Central Legitimization Agency* (CLA) dan *Central Tabulating Facilities* (CTF).

Selain itu aspek keamanan pada *Two Central Facilities* dapat dicapai dengan menggunakan algoritma kriptografi, menerapkan konsep dengan menyembunyikan informasi, dan menerapkan protokol keamanan. Kriptografi digunakan untuk menjaga pesan dari pihak yang tidak memiliki hak untuk mengakses suatu informasi. Kerahasiaan dapat dicapai dengan menggunakan ukuran keamanan algoritma fisik atau matematika (Firdaus, Wahyudin, & Nugroho, 2017).

Salah satu alasan algoritma RSA (*Rivest—Shamir—Adleman*) paling banyak digunakan untuk kriptografi adalah karena memungkinkan salah satu dari dua kunci untuk mengenkripsi pesan dan kunci yang berlawanan untuk mendekripsi, sehingga menjanjikan kerahasiaan, integritas, keaslian, dan non-reputasi data dan komunikasi elektronik (Nisha & Farik, 2017). Dalam algoritma RSA, satu pihak menggunakan kunci publik dan pihak lain menggunakan kunci rahasia, yang dikenal sebagai kunci pribadi (Saranya, Vinothini, & Vasumathi, 2014).

2. METODE PENELITIAN

Penelitian ini menggunakan metode *Security Life Cycle*. Terdapat-tahapan tahapan utama yang diciptakan dalam *Security Life Cycle* (Bishop, 2003).



Gambar 1. Security life cycle

1) Ancaman (*Threats*)
Sistem *Computer Based Test* diharapkan mampu melindungi sistem dari berbagai ancaman yang mungkin terjadi. Contoh dari ancaman ini antara lain :
   - Modifikasi identitas
     Yakni ancaman perubahan data-data yang ada di server seperti pengubahan nama, NISN dan data diri lainnya
   - Penyamaran
     Yakni ancaman yang berupa peniruan suatu entitas terhadap entitas yang lain, entitas disini sebagai peserta ujian.

- Disruption
  Yakni penyerangan terhadap sistem, penyerangan ini melemahkan sumber daya sistem sehingga tidak dapat diakses atau sistem mengalami crash. (Muharram & Satrya, 2015)

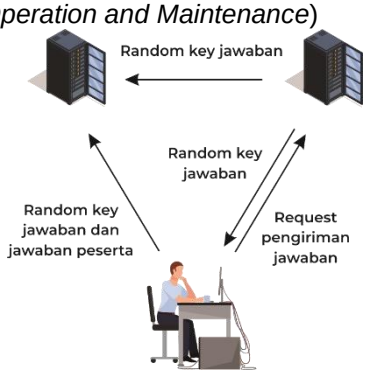
2) Kebijakan (*Policy*)
Keamanan CBT ini dibagun secara komputerisasi dan dapat digunakan apabila terdapat dua aturan yakni privasi siswa dan pencegahan terhadap kecurangan. Dalam suatu aturan yang aman harus memiliki beberapa persyaratan antara lain:
   - Hanya peserta ujian yang dapat mengerjakan ujian CBT atau login atau autentikasi
   - Peserta tidak bisa memilih lebih dari satu jawaban
   - Peserta tidak boleh menggantikan atau digantikan oleh orang lain.
   - Peserta boleh memastikan kembali identitas atau jawaban yang terisi sebelum selesai ujian.

3) Spesifikasi (*Specification*)
Pada sistem ini terdiri dari dua server yaitu server CLA, server CTF. Server CLA akan mengautentikasi siswa menggunakan NISN dan password. Sedangkan server CTF menyediakan soal yang akan dikerjakan oleh peserta ujian. CTF bertindak dalam pengumpulan ja\waban peserta ujian dan menghitung jawaban yang benar. Seluruh data akan tersimpan secara cipherteks menggunakan algoritma RSA, sehingga keamanan informasi lebih terjaga. Secara umum, sistem yang dibangun haruslah memberikan jaminan bahwa informasi yang diakses pengguna tidak diganggu oleh pihak pihak yang tidak berwenang dalam mengakses sistem, namun tetap mempertimbangkan sisi kecepatan pertukaran data. Pengiriman data dalam setiap proses, misalnya registrasi juga haruslah terjamin keamanannya sehingga diperlukan pengenkripsian data sebelum pengiriman dilakukan.

4) Perancangan (*Design*)
Pada sistem ini dibagi menjadi dua bagian, yaitu perancangan sistem secara umum yang membahas keseluruhan sistem yang dibagun menggunakan protokol *Two Central Facilities* yang telah dimodifikasi. Dalam melakukan perancangan server CLA dan CTF, serta penggunaan algoritma RSA perlu diketahui bahwa penggunaan setiap server. Selain itu perlu penentuan hardware yang sesuai dengan kebutuhan sistem.

5) Implementasi (*Implementation*)
Sistem CBT idealnya adalah suatu perangkat yang dirakit untuk menjadi sistem ujian berbasis online atau yang biasa dikenal dengan CBT dengan kebijakan sebagai berikut :
   - Peserta atau siswa disediakan keyboard, mouse dan headphone untuk memudahkan siswa mengerjakan soal-soal yang diujikan.
   - Siswa hanya berinteraksi dengan sistem CBT menggunakan perangkat yang telah disediakan panitia ujian.

6) Operasi dan Pemeliharaan (*Operation and Maintenance*)



Gambar 2. Skema Protokol Two Central Facilities

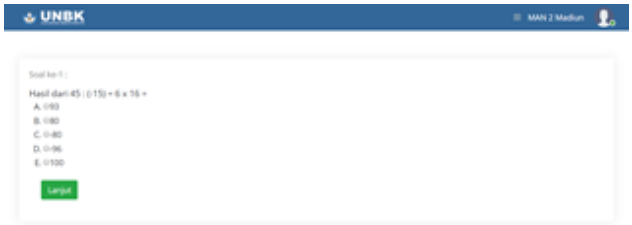
3. HASIL DAN PEMBAHASAN

4. Hasil Implementasi

Implementasi *Two Central Facilities* pada UNBK yaitu dalam penggunaan *Central Legitimization Agency* (CLA) sebagai autentikasi siswa dan *Central Tabulating Facilities* (CTF) sebagai perhitungan hasil jawaban peserta. Pada sistem ini terdapat tiga komponen diantaranya tampilan user, server CLA dan server CTF. Tampilan user merupakan komponen untuk peserta agar dapat berinteraksi dengan sistem, sehingga peserta dapat mengerjakan soal-soal CBT. CLA merupakan pusat data yang berfungsi dalam penyimpanan data siswa. Data tersebut tidak dapat dilihat orang lain termasuk pihak *Central Tabulating Facilities* (CTF). Data yang ada di *Central Legitimization Agency* (CLA) digunakan untuk login dan autentikasi. Sedangkan *Central Tabulating Facilities* (CTF) merupakan komponen kedua yang ada di sistem ini dan berfungsi untuk penyedia soal dan perhitungan nilai dari jawaban siswa.

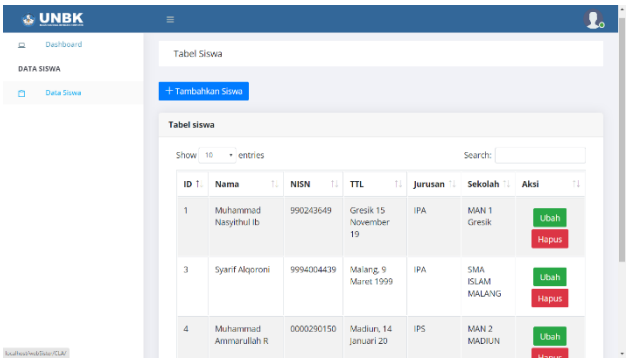
Gambar 3. Halaman autentikasi siswa

Gambar 3 merupakan tampilan di halaman pertama yakni form login peserta ujian UNBK. Gambar tersebut mempunyai inputan User ID berupa NISN dan password, kedua form tersebut sebagai fungsi autentikasi peserta di sistem CBT.



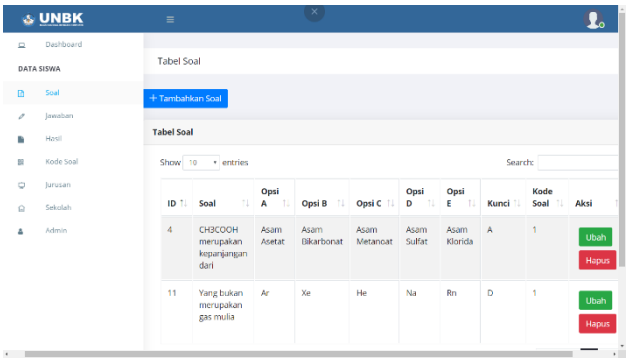
Gambar 4. Tampilan UNBK

Gambar 4 merupakan tampilan UNBK yang berisi tentang soal dan opsi jawaban yang akan dikerjakan oleh peserta ujian. Soal yang telah dikerjakan semuanya akan dikirimkan ke server CTF yang akan dihitung hasil ujiannya.

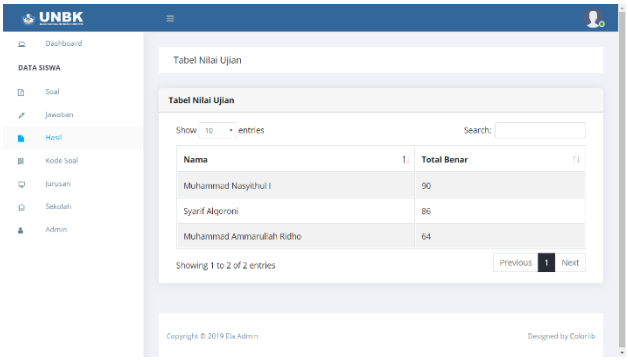


Gambar 5. Tampilan Data Siswa pada CLA

Tampilan dashboard pada Gambar 5 digunakan admin untuk menginputkan data siswa sebagai peserta ujian. Data yang diinputkan berupa nama, NISN, TTL, jurusan, dan asal sekolah.

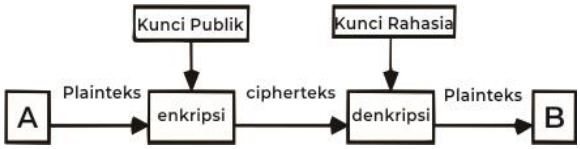


Gambar 6. **Tampilan Data Soal pada CTF**



Gambar 7. **Tampilan Hasil penilaian pada CTF**

Gambar 6 merupakan tampilan tabel soal yang digunakan admin untuk menginputkan data soal beserta kunci jawaban. Sedangkan pada Gambar 7 merupakan hasil yang telah dilakukan setelah pencocokan jawaban siswa dengan kunci jawaban.



Gambar 8. Skema algoritma RSA

Gambar 8 menjelaskan tentang proses enkripsi dengan algoritma RSA menggunakan sebuah kunci publik sehingga menghasilkan suatu chiperteks. Sedangkan proses dekripsi menggunakan kunci rahasia yang disediakan oleh server.

5. Uji Fungsionalitas
Uji fungsional diterapkan pada manajemen data siswa pada CLA, manajemen data soal pada CTF dan tampilan pengerjaan ujian.

Tabel 1. Hasil pengujian fungsional sistem

Skenario Pengujian	Hasil
Penginputan data siswa pada server CLA	Berhasil
Semua data telah terenkripsi pada server CLA	Berhasil
Penginputan data siswa pada server CTF	Berhasil
Semua data telah terenkripsi pada server CTF	Berhasil
Perangkat ujian mengambil data dari CLA dan CTF	Berhasil
Server CTF dapat menghitung hasil ujian peserta	Berhasil

Berdasarkan Tabel 1, perangkat ujian dapat mengambil data siswa sebagai autentikasi user dari server CLA dan mengambil data soal pada server CTF serta mengirimkan hasil jawaban ke server CTF. Server CTF telah berhasil dalam perhitungan hasil ujian.

Gambar 9. Data yang telah dienkripsi dengan Algoritma RSA

Berdasarkan Gambar 9, perangkat ujian CBT telah berhasil terenkripsi menggunakan Algoritma RSA, sehingga keamaan data lebih terjaga. Berdasarkan hasil uji keamanan, chiperteks tidak bisa didekripsi dengan kunci yang tidak sesuai dengan ketentuan yang ada.

KESIMPULAN

Setelah dilakukanya pengembangan terhadap aplikasi *Computer Based Test* dengan Protokol *Two Central Facilities* yang menerapkan sistem *Central Legitimization Agency* (CLA) dan *Central Tabulating Facilities* (CTF) telah berhasil menciptakan sistem CBT yang terjaga keamanan datanya. Ditambah lagi adanya enkripsi pada data dengan Algoritma RSA yang dapat membuat keamanan data lebih terjaga. Dengan adanya penerapan Algoritma RSA, data yang tersimpan pada sistem berupa chiperteks. Chiperteks hanya bisa didekripsi menggunakan kunci yang telah ditentukan oleh sistem, sehingga selain kunci yang ditentukan data tidak bisa terbaca oleh siapapun.

DAFTAR PUSTAKA

Bishop, M. (2003). Computer Security Art and Science. *Pearson Education, Inc. Boston*.

Firdaus, C., Wahyudin, & Nugroho, E. P. (2017). Monitoring System with Two Central Facilities Protocol. *Jurnal UPI*, 1.

Kemendikbud. (2019, November 19). *Tentang UNBK*. Retrieved from Kemendikbud UNBK: https://ubk.kemdikbud.go.id/

Muharram, A. T., & Satrya, F. (2015). Rancang Bangun Sistem E-Voting Menggunakan Protokol Two Central Facilities. *Jurnal Informatik*, 37.

Nisha, S., & Farik, M. (2017). RSA Public Key Cryptography Algorithm – A Review. *IJSTR*, 187.

Poggio, J., Glassnapp, D., & Yang, X. (2005). A Com-parative Evaluation of Score Results from Com-puterized and Paper & Pencil Mathematics Test-ing in a Large Scale State Assessment Program. *The Journal of Technology, Learning, and As-sessment*, 4-30.

Saranya, Vinothini, & Vasumathi. (2014). A Study on RSA Algorithm for Cryptography. *IJCSIT*, 5708.

Schneir, B. (1996). Applied Cryptography. *Ed ke-2, Jon Wiley & Sons*.

Sireesha, J., & Chakchai, S.-I. (2005). *Secure Virtual Election Booth with Two Central Facilities.* Washington: Department of Computer Science Washington University.

JISKa, Vol. 4, No. 3, JANUARI, 2020, Pp. 156 – 164
ISSN : 2527 – 5836 (print) | 2528 – 0074 (online)

Deteksi Serangan *Distributed Denial of Services* (DDOS) Berbasis HTTP Menggunakan Metode *Fuzzy* Sugeno

Nadila Sugianti [(1)], Yayang Galuh [(2)], Salma Fatia [(3)], Khadijah Fahmi Hayati Holle [(4)]
[(1,2,3,4)]Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang
Jl. Gajayana No.50, Dinoyo, Kec. Lowokwaru, Kota Malang, Jawa Timur 65144
*Email: nadilasgnti@gmail.com[(1)], yayanggaluh1198@gmail.com[(2)], salmafatia7@gmail.com[(3)], khadijah.holle@uin-malang.ac.id[(4)]

Abstract

Distributed Denial of Services (DDOS) is a type of attack that exploits the web. This attack causes the server to go down and a system error. Thus, early detection of DDOS attacks is fundamental. The purpose of this paper is to develop applications that are capable of detecting HTTP-based DDOS attacks. This paper uses the sugeno fuzzy method for a systematic approach. From several studies that have been conducted, the researchers identified that the optimal input variables included the number of users, number of packages, number of lengths / users, and length of packages. Data processing used MATLAB software. The validity of the test uses the formula for the level of accuracy as in equation (3), resulting in an application that is able to detect HTTP-based DDOS attacks using sugeno fuzzy method with an accuracy rate of up to 90%.

Keywords : DDOS, Fuzzy Logic, Fuzzy Sugeno, HTTP

Abstrak

Distributed Denial of Services (DDOS) merupakan salah satu jenis serangan yang mengeksploitasi web. Serangan ini mengakibatkan server menjadi down dan system error. Sehingga, deteksi dini serangan DDOS merupakan hal yang fundamental. Tujuan paper ini adalah membangun aplikasi yang mampu mendeteksi serangan DDOS berbasis HTTP. Paper ini menggunakan metode fuzzy sugeno untuk pendekatan sistematis. Dari beberapa penelitian yang sudah dilakukan, peneliti mengidentifikasi bahwa variabel input yang optimal meliputi jumlah user, jumlah paket, jumlah panjang/user, dan panjang paket. Pengolahan data digunakan software MATLAB. Validitas pengujian menggunakan rumus tingkat keakuratan seperti pada Pers.(3), menghasilkan sebuah aplikasi yang mampu mendeteksi serangan DDOS berbasis HTTP menggunakan metode fuzzy sugeno dengan tingkat keakuratan mencapai 90%.

Kata Kunci : ***DDOS, Logika Fuzzy, Fuzzy Sugeno, HTTP***

1. PENDAHULUAN

Website atau yang disingkat *web* merupakan sistem yang di dalamnya terdapat informasi berupa manuskrip, gambar ataupun suara yang umumnya ditulis memakai format HTML (Sugianto, 2003). Untuk dapat mengakses HTML, diperlukan sebuah protokol bernama *Hypertext Transfer Protocol* (HTTP). *Distributed Denial of Services* (DDOS) merupakan salah satu jenis eksploitasi kelemahan pada sebuah *web*. DDOS merupakan upaya serangan terhadap *server* di dalam jaringan internet dengan cara membanjiri banyak data pada lalu lintas jaringan untuk mengganggu jalannya lalu lintas normal pada *server*. Serangan ini dapat mengakibatkan *server* menjadi *down* dan mengakibatkan *system error* (Adrian & Isnianto, 2016). Oleh karena itu, deteksi dini serangan DDOS merupakan hal yang fundamental.

Serangan DDOS dapat dideteksi menggunakan teknik *soft computing*, salah satunya adalah *fuzzy logic*. Metode fuzzy sugeno diusulkan sebagai pendekatan sistematis karena mampu memberikan representasi yang lebih efisien secara komputasi jika dibandingkan mamdani. Dengan menggunakan pendekatan sugeno jumlah aturan jauh lebih kecil daripada metode *fuzzy* mamdani. Metode sugeno terlah berhasil diterapkan pada sejumlah masalah dunia nyata seperti perkiraan fungsi non-linear statis, seperti prediksi pasar saham (Kulkolj, 2002).

Beberapa penelitian terdahulu sudah membahas metode *fuzzy* sugeno untuk mendeteksi serangan DDOS, namun beberapa variabel tidak disertakan (Petkovic, Basicevic, Kukolj, & Popovic, 2015). Pada paper ini peneliti menentukan variabel *input* yang optimal untuk metode sugeno agar mencapai tingkat deteksi yang lebih baik dalam lingkungan yang dinamis (Petkovic, Basicevic, Kukolj, & Popovic, 2015). Paper ini diharapkan dapat menciptakan suatu aplikasi guna mendeteksi serangan DDOS berbasis HTTP dengan tingkat akurasi yang baik menggunakan metode *fuzzy* sugeno.

2. METODE PENELITIAN

Mekanisme penelitian ialah semua proses yang dibutuhkan mulai dari perencanaan hingga pelaksanaan penelitian. Mekanisme penelitian yang diimplementasikan terbagi ke dalam beberapa tahapan.

2.1 Pengambilan Data Awal

Informasi penelitian yang digunakan berupa data sekunder yang terdiri dari data variabel yaitu, jumlah user, jumlah paket, jumlah paket/user dan panjang paket. Setiap variabel terdiri dari status yang normal atau DDOS. Data ini di dapatkan dengan melakukan 10 kali percobaan pengambilan paket pada website menggunakan *Wireshark* selama delapan detik. Dari data yang sudah di dapatkan, kemudian dimasukkan ke dalam Microsoft Excel. Data hasil deteksi terdapat pada Tabel 1.

Tabel 1. Hasil Deteksi Website E-Learning UIN Malang.

No	User	Jumlah Paket	Panjang/User	Panjang Paket	Status
1	3	32	228.9583	686.875	Normal
2	3	287	291.2114	873.6341	Normal
3	3	319	270.0491	810.1473	Normal
4	3	377	275.2909	825.8727	Normal
5	3	364	281.2363	843.7088	Normal
6	3	13943	230.5874	691.7621	DDOS
7	3	19818	245.2742	735.8226	DDOS
8	3	20214	244.811	734.4329	DDOS
9	3	26193	251.1058	753.3173	DDOS
10	3	17748	246.7402	740.2205	DDOS

2.2 Penentuan Fungsi Keanggotaan Himpunan *Fuzzy*

Himpunan *fuzzy* (*fuzzy set*) merupakan sebuah pengelompokkan yang dinyatakan dalam fungsi keanggotaan (*membership function*). Definisi fungsi keanggotaan yaitu, grafik yang menunjukkan batas nilai input suatu variabel dengan interval nilai antara 0 sampai 1 (Kusumadewi & Purnomo, 2010).

Terdapat beberapa fungsi keanggotaan himpunan *fuzzy* seperti fungsi keanggotaan linear, fungsi keanggotaan segitiga, dan fungsi keanggotaan trapesium. Peneliti menggunakan fungsi keanggotaan segitiga seperti pada Pers.(1)

$$\mu(x) = \begin{cases} 0 & ; x \leq \text{a atau } x \geq \text{c} \\ \frac{(x-a)}{(b-a)} & ; \text{a} \leq x \leq \text{b} \\ \frac{(c-x)}{(c-b)} & ; \text{b} \leq x \leq \text{c} \end{cases} \quad (1)$$

2.2 Pengukuran Nilai Penegasan (Defuzzifikasi)

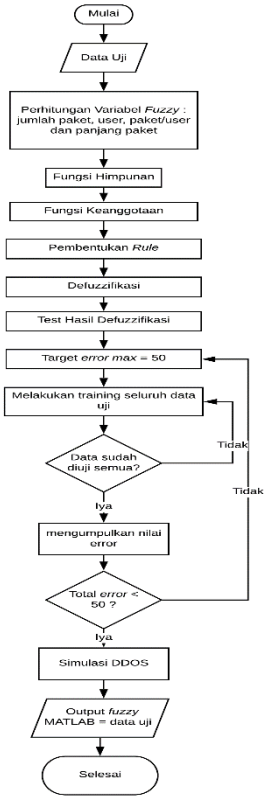
Nilai defuzzifikasi didapatkan dengan cara mencari nilai rata-ratanya. Rumus defuzzifikasi dituliskan sebagaimana pada Pers.(2) berikut.

$$Z = \frac{\sum_{i=1}^{n} a_i z_i}{\sum_{i=1}^{n} a_i} \qquad (2)$$

Nilai dimana α_i merupakan α predikat ke-*i*, sedangkan z_i merupakan hasil pada anteseden aturan ke-*i*.

2.2 Perancangan Aplikasi Pendeteksi DDOS

Aplikasi pendeteksi DDOS ini dibuat menggunakan pemrograman MATLAB R2017B dengan menggunakan *tools* pemrogramannya yang bernama *fuzzy toolbox.* Aplikasi dibangun menggunakan *interface* GUI MATLAB sehingga tampilan menjadi menarik dan lebih mudah untuk digunakan. Gambar 1 merupakan alur dari perancangan aplikasi DDOS.



Gambar 1. Alur Aplikasi DDOS.

3. HASIL DAN PEMBAHASAN

Dalam sistem pendeteksi serangan DDOS ini, terdapat empat variabel input, yaitu jumlah user, jumlah paket, panjang paket, dan jumlah panjang dibagi dengan user. Sedang, variabel output-nya adalah status. Penentuan variabel semesta pembicaraan terdapat pada Tabel 2.

Tabel 2. Semesta Pembicaraan Variabel Input dan Output.

Fungsi	Variabel	Semesta Pembicaraan

Input	Jumlah user	[0 - 7]
	Jumlah paket	[0 – 29.000]
	Jumlah panjang/user	[0 - 292]
	Panjang paket	[0 - 875]
Output	Status	[0 - 1]

Pada tabel 2, yang menjadi semesta pembicaraan merupakan data minimal dan data maksimal setiap variabel yang didapat dari fungsi keanggotaan himpunan *fuzzy*. Tahapan membuat fungsi keanggotaan sesuai pada Pers.(1) dilakukan untuk mencari nilai domain.

Fungsi keanggotaan himpunan fuzzy pada variabel Jumlah user terdiri dari SEDIKIT, SEDANG, dan BANYAK.

$$\mu\text{Sedikit} = \begin{cases} 0 & ; x \leq 0 \text{ atau } x \geq 3 \\ \frac{(x-1)}{(1{,}5-0)} & ; 0 \leq x \leq 1{,}5 \\ \frac{(1{,}5-x)}{(3-1{,}5)} & ; 1{,}5 \leq x \leq 3 \end{cases}$$

$$\mu\text{Sedang} = \begin{cases} 0 & ; x \leq 2 \text{ atau } x \geq 5 \\ \frac{(x-2)}{(3{,}5-2)} & ; 2 \leq x \leq 3{,}5 \\ \frac{(3{,}5-x)}{(5-3{,}5)} & ; 3{,}5 \leq x \leq 5 \end{cases}$$

$$\mu\text{Banyak} = \begin{cases} 0 & ; x \leq 4 \text{ atau } x \geq 7 \\ \frac{(x-4)}{(5{,}5-4)} & ; 4 \leq x \leq 5{,}5 \\ \frac{(5{,}5-x)}{(7-5{,}5)} & ; 5{,}5 \leq x \leq 7 \end{cases}$$

Fungsi keanggotaan himpunan fuzzy pada variabel Jumlah data terdiri dari SEDIKIT, SEDANG, dan BANYAK.

$$\mu\text{Sedikit} = \begin{cases} 0 & ; x \leq 0 \text{ atau } x \geq 1000 \\ \frac{(x-1)}{(500-0)} & ; 0 \leq x \leq 1000 \\ \frac{(500-x)}{(1000-500)} & ; 500 \leq x \leq 1000 \end{cases}$$

$$\mu\text{Sedang} = \begin{cases} 0 & ; x \leq 500 \text{ atau } x \geq 10.000 \\ \frac{(x-500)}{(5.000-500)} & ; 500 \leq x \leq 5.000 \\ \frac{(5.000-x)}{(10.000-5.000)} & ; 5.000 \leq x \leq 10.000 \end{cases}$$

$$\mu\text{Banyak} = \begin{cases} 0 & ; x \leq 5.000 \text{ atau } x \geq 29.000 \\ \frac{(x-5.000)}{(14.000-5.000)} & ; 5.000 \leq x \leq 14.000 \\ \frac{(14.000-x)}{(29.000-14.000)} & ; 14.000 \leq x \leq 29.000 \end{cases}$$

Fungsi keanggotaan himpunan fuzzy pada variabel Jumlah paket/user terdiri dari KECIL, dan BESAR.

$$\mu\text{Kecil} = \begin{cases} 0 & ; x \leq 0 \text{ atau } x \geq 200 \\ \frac{(x-0)}{(100-0)} & ; 0 \leq x \leq 100 \\ \frac{(100-x)}{(200-100)} & ; 100 \leq x \leq 200 \end{cases}$$

$$\mu\text{Besar} = \begin{cases} 0 & ; x \leq 100 \text{ atau } x \geq 196 \\ \frac{(x-100)}{(196-100)} & ; 100 \leq x \leq 196 \\ \frac{(196-x)}{(292-196)} & ; 196 \leq x \leq 292 \end{cases}$$

Fungsi keanggotaan himpunan fuzzy pada variabel Panjang data terdiri dari KECIL, dan BESAR.

$$\mu\text{Kecil} = \begin{cases} 0 & ; x \leq 0 \text{ atau } x \geq 610 \\ \frac{(x-0)}{(305-0)} & ; 0 \leq x \leq 305 \\ \frac{(305-x)}{(610-305)} & ; 305 \leq x \leq 610 \end{cases}$$

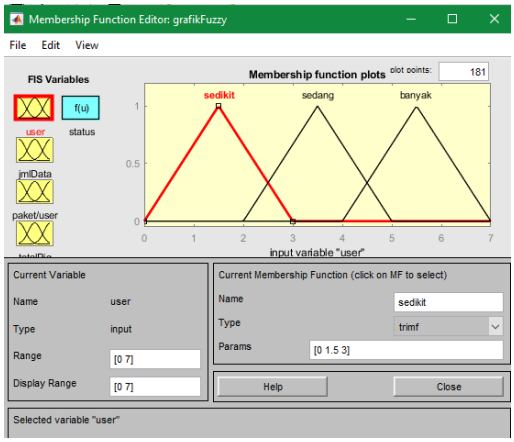
$$\mu\text{Besar} = \begin{cases} 0 & ; x \leq 305 \text{ atau } x \geq 875 \\ \frac{(x-305)}{(570-305)} & ; 305 \leq x \leq 570 \\ \frac{(570-x)}{(875-570)} & ; 570 \leq x \leq 875 \end{cases}$$

Berdasarkan fungsi keanggotaan dari setiap variabel *input* yang telah dijelaskan, maka diperoleh nilai domain untuk setiap himpunan *fuzzy*. Nilai domain terdapat pada Tabel 3.

Tabel 3. Nilai Domain pada Variabel

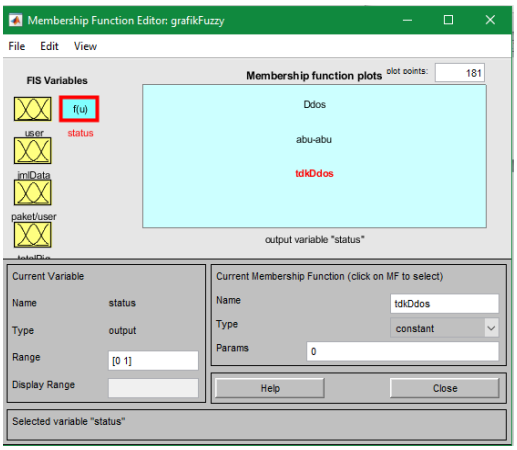
Fungsi	Variabel	Himpunan	Nilai Domain
Input	Jumlah user	Sedikit	[0 - 3]
		Sedang	[2 - 5]
		Banyak	[4 - 7]
	Jumlah paket	Sedikit	[0 – 1.000]
		Sedang	[500 –10.000]
		Banyak	[5.000 – 29.000]
	Jumlah panjang/user	Kecil	[0 - 200]
		Besar	[100 - 292]
	Panjang paket	Kecil	[0 - 610]
		Besar	[305 - 875]
Output	Status	Normal	[0]
		DDOS Ringan	[0.6]
		DDOS Berat	[0.9]

Selanjutnya melakukan implementasi menggunakan *software* bantuan berupa MATLAB. Gambar 2 merupakan bentuk dari fungsi keanggotaan himpunan *fuzzy* pada variabel input jumlah user dengan menggunakan *fuzzy toolbox* pada MATLAB.



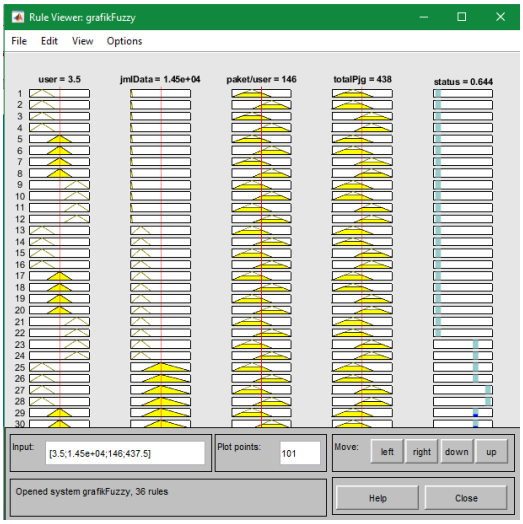
Gambar 2. Fungsi Keanggotaan Variabel *User* pada MATLAB.

Pembentukan himpunan *fuzzy* untuk variabel *output* terdapat pada Gambar 3 dengan tipe fungsi keanggotannya berupa *constant*.



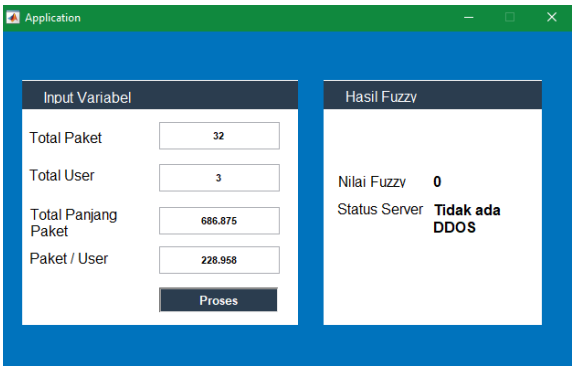
Gambar 3. Fungsi Keanggotaan Variabel *Output* Status.

Tahapan pembentukan aturan *fuzzy*. Terdapat 36 aturan *fuzzy* yang diimplementasikan ke dalam sistem pendeteksi DDOS ini. Pembentukan aturan *fuzzy* didapatkan dari empat variabel *input* dan satu variabel *output* yang sudah di definisikan sebelumnya dengan melakukan analisa batas tiap-tiap himpunan. Gambar 4 merupakan hasil pembentukan aturan *fuzzy* (*rule view*) berdasarkan variabel linguistik dan variabel numerik pada *fuzzy toolbox* MATLAB.



Gambar 4. Rule View (Hasil Optimasi/Defuzzifikasi).

Pembuatan model *fuzzy* untuk mendeteksi serangan DDOS berbasis HTTP akan dikatakan baik jika sudah dilakukan pengujian. Peneliti menggunakan 10 sampel untuk mengetahui keakuratan dan *error* pada aplikasi pendeteksi DDOS yang kami bangun berdasarkan hasil pengujian. Pada bagian ini dilakukan pengujian data berdasarkan paket data yang telah didapatkan pada Tabel 1.



Gambar 5. Tampilan Hasil Pengujian Data 1.

Gambar 5 merupakan tampilan dari aplikasi yang dibangun menggunakan GUI MATLAB. Aplikasi dijalankan untuk melakukan pengujian seluruh sampel. Tabel 4 merupakan hasil perolehan pengujian data yang dilakukan.

Tabel 4. Hasil Pengujian Sampel dengan MATLAB.

No	Data Uji	Matlab	Hasil
1	Normal	Normal	Sesuai
2	Normal	Normal	Sesuai
3	Normal	Normal	Sesuai
4	Normal	Normal	Sesuai
5	Normal	Normal	Sesuai
6	DDOS	Normal	Tidak Sesuai

7	DDOS	DDOS	Sesuai
8	DDOS	DDOS	Sesuai
9	DDOS	DDOS	Sesuai
10	DDOS	DDOS	Sesuai

Pada tabel 4, didapatkan perbandingan nilai logika *fuzzy* sugeno untuk mendeteksi serangan DDOS berbasis HTTP. Pengujian ini memakai rumus tingkat keakuratan.

$$\text{Tingkat Keakuratan} = \frac{Jumlah\ Data\ Benar}{Jumlah\ Data\ Keseluruhan}\ x\ 100\% \qquad (3)$$

Berdasarkan Tabel 4 diatas, dari 10 data yang digunakan sebagai sampel dalam penelitian ini 9 diantaranya dapat terdeteksi secara akurat. 10% dari data sampel tidak dapat dideteksi secara akurat karena pada data acuan yang dipakai memiliki konstanta yang telah ditentukan sebagai pembeda antara data normal, DDOS ringan, dan DDOS berat. Sedangkan, data tersebut masih termasuk ke dalam limit data yang normal. Sehingga tingkat ketepatan dari model fuzzy ini mencapai 90% dengan *error* 10%. Dengan demikian, logika fuzzy yang menggunakan metode sugeno telah memenuhi batasan dari paper ini yaitu untuk mendeteksi serangan DDOS berbasis HTTP.

4. KESIMPULAN

Berdasarkan pembahasan yang telah dijelaskan dan hasil pengujian yang telah dilakukan mengenai masalah mendeteksi serangan DDOS berbasis HTTP berdasarkan jumlah user, jumlah paket, jumlah paket/user dan panjang data yang ditangkap maka dapat diambil kesimpulan, yaitu :

1. Untuk mendeteksi suatu website terkena serangan atau tidak dapat memasukkan nilai pada kolom input yang terdapat pada Gambar 5 sesuai dengan data yang sudah di dapatkan.
2. Pada hasil pengujian, logika fuzzy menggunakan metode sugeno mampu digunakan sebagai pendeteksi dalam menentukan serangan DDOS berbasis HTTP dengan tingkat keakuratan mencapai 90%.

DAFTAR PUSTAKA

Adrian, R., & Isnianto, H. N. (2016). Analisa Pengaruh Variasi Serangan DDOS pada Performa Router. *Seminar Nasional Teknologi Terapan*.

Kulkolj, D. (2002). Design of Adaptive Takagi-Sugeno-Kang Fuzzy Model. *Applied Soft Computing*, 89-103.

Kusumadewi, S., & Purnomo, H. (2010). *Aplikasi Logika Fuzzy untuk Pendukung Keputusan.* Yogyakarta: Graha Ilmu.

Muhammad, A. W., & Alameka, F. (2017). Integrasi Normalized Relative Network Entropy dan Neural Network Backpropagation (BP). *JURTI*, 1-6.

Muhammad, A. W., Riadi, I., & Sunardi. (2016). Analisis Statistika Log Jaringan untuk Deteksi Serangan DDOS Berbasis Neural Network. *Jurnal Ilmiah ILKOM*.

Petkovic, M., Basicevic, I., Kukolj, D., & Popovic, M. (2015). Evaluation of Takagi-Sugeno-Kang Fuzzy Method in Entropy-based Detection of DDoS Attacks. *Computer Science and Information Systems*, 139-162.

Rahakbauw, D. L. (2015). Penerapan Logika Fuzzy Metode Sugeno untuk Menentukan Jumlah Produksi Roti Berdasarkan Data Persediaan dan Jumlah Pertanian. *Jurnal Ilmu Matematika dan Ilmu Terapan*, 121-134.

Rumare, R. R., Ciptaningtyas, H. T., & Santoso, B. J. (2017). Aplikasi Pendeteksi Serangan pada HTTP Menggunakan N-Gram. *Jurnal Teknik ITS*.

Sihombing, R. O., & Zulfin, M. (n.d.). Analisis Kinerja Trafik Web Browser dengan Wireshark Network Protocol Analyzer pada Sistem Client-Server.

Simanjuntak, P., Suharyanto, C. E., & Khairiyah, R. (2018). Fuzzy Sugeno untuk Menentukan Penilaian kompetensi Karyawan PT. Scheinder Batam. *Information Sytem Developtment*, 2.

Siregar, J. J. (n.d.). Analisis Exploitasi Keamanan Web Denial of Services Attack.

Sugianto, D. (2003). *LDL Membangun Website dengan PHP.* Jakarta: D@takom.

Wardhana, L., & Makodian, N. (2010). *Teknologi Wireless COmmunication dan Wireless Broadcast.* Jakarta: Andi Offset.

JISKa, Vol. 4, No. 3, JANUARI, 2020, Pp. 165 – 172
ISSN : 2527 – 5836 (print) | 2528 – 0074 (online)

DESAIN DAN IMPLEMENTASI SIMULASI *INTRUSION INDEX* BERBASIS SISTEM PAKAR DENGAN METODE *FORWARD CHAINING*

Mardian [(1)], H. Jemakmun [(2)], Linda Atika [(3)]
Program Pascasarjana Universitas Bina Darma
Jl. Jenderal Ahmad Yani No.3, 9/10 Ulu, Kecamatan Seberang Ulu I, Kota Palembang, Sumatera Selatan 30111
e-mail : mardiannelwan@gmail.com [(1)], jemakmun@binadarma.ac.id [(2)], linda.atika@binadarma.ac.id [(3)]

Abstract

Today's internet needs are increasing, the interest and attention on the internet is also getting wider and faster on the internet network, especially from disruption of attacks or illegal access to the internet network itself. Network ssecurity depends on the speed of nework setting in the case or following up the system when an interruption occurs. For this reason, software is needed that is capable of detecting and measurement attacks using an expert system. The results of the design and simulation carried out in this study can illustrate the way or flow of the use of the system from the manager who becomes an actor, to shows how the flow of information flows in a system and can provide a static picture of the system that shows relationships or interconnected relationships between classes related to information systems expert system for internet network security and the application of intrusion index to dassify the types of attacks into three levels, namely Deflect, Prevent, and Preempt by applying inference engine into Forward Chaining method.

Keywords**: **expert systems, network security, forward chaining method

Abstrak

Adanya kebutuhan internet yang semakin meningkat, ketertarikan dan perhatian pada internet juga semakin luas dan cepat sehingga harus diseimbangi dengan keamanan yang lebih cepat pada jaringan internet, terutama dari hal gangguan serangan atau akses ilegal pada jaringan internet. Keamanan jaringan bergantung pada kecepatan pengaturan jaringan dalam hal menindaklanjuti sistem saat terjadi gangguan. Untuk itu diperlukan suatu perangkat lunak yang mampu melakukan deteksi dan pengukuran serangan dengan menggunakan sistem pakar. Hasil dari desain dan simulasi yang dilakukan dalam penelitian ini dapat menggambarkan cara atau alur penggunaan sistem dari pengelola yang menjadi aktor, memperlihatkan cara aliran informasi mengalir dalam suatu sistem serta dapat memberikan gambaran sistem secara statis yang memperlihatkan relasi atau hubungan antarkelas yang saling berkaitan mengenai sistem informasi sistem pakar untuk keamanan jaringan internet dan penerapan *intrusion index* yang dapat menggolongkan jenis serangan ke dalam tiga tingkatan yaitu *deflect*, *prevent*, dan *preempt* dengan menerapkan mesin inferensi ke dalam metode *forward chaining*.

Kata Kunci : sistem pakar, keamanan jaringan, metode *forward chaining, Intrusion Index*

1. PENDAHULUAN

Adanya kebutuhan internet yang semakin meningkat, ketertarikan dan perhatian pada internet juga semakin luas dan cepat sehingga harus diseimbangi dengan keamanan yang lebih cepat pada jaringan internet, terutama dari hal gangguan serangan atau akses ilegal pada jaringan internet. Keamanan jaringan bergantung pada kecepatan pengatur jaringan dalam hal menindaklanjuti sistem saat terjadi gangguan. Pada penelitian sebelumnya menurut (Sodiya, Adeniran, & Ikuomola, 2007) , Ada 3 (tiga) jenis golongan atau rule yaitu 1. *Deflect* Jika terdapat serangan di dalam database 2. *Prevent* Jika terdapat gangguan di jaringan. 3. *Preempt* Jika terdapat gangguan yang lebih parah, baik di database maupun di jaringan yang sifatnya

berlanjut. *Intrusion Index* adalah suatu proses yang digunakan untuk mengukur dan mengetahui bahaya serangan yang dilakukan oleh seseorang ketika gangguan terdektesi.

Untuk menganalisa hal tersebut maka diperlukan sebuah pengukuran jenis golongan serangan yang terjadi dan analisa struktur basis data sistem pakar agar mudah digunakan oleh seorang administrator secara efektif sehingga penelitian bisa mengetahui sistem yang mampu memberikan respon secara cepat terhadap sistem pemberi peringatan seperti IDS sehingga seorang administrator dapat mengetahui apa pengertian arti peringatan tersebut dan bagaimana mengefektifkan respon tersebut. sehingga penelitian bisa mengetahui keefektifan metode *forward chaining* dalam memproses *Intrusion Index*.

Sistem pakar akan menjadi layaknya seorang pakar didalam bidang tertentu sesuai kebutuhan manusia. Sistem pakar mampu memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dengan menggunakan pengetahuan, fakta dan teknik penalaran (Desiani, 2006)

Salah satu cara representasi *knowledge* (pengetahuan) adalah melalui *rule*. *Rule* merupakan struktur *IF/THEN* yang secara logika menghubungkan informasi yang tersimpan dalam bagian *IF* yang juga dikenal sebagai premis, dengan informasi yang tersimpan dalam bagian *THEN*. Kumpulan *rule* yang saling terkait disebut juga sebagai *rule set*. Bentuk umum *rule* sebagai berikut (Turban, 1995)

1. METODOLOGI PENELITIAN

Langkah pertama dalam penelitian ini adalah pengumpulan data dan analisis kebutuhan. Sumber pengumpulan data adalah penelitilan sebelumnya dan literature yang bersumber dari internet. Langkah kedua adalah mendesain struktur basis data. Langkah ketiga adalah pengukuran *Intrusion Index* dimana digunakan untuk mengukur dan mengetahui bahaya serangan yang dilakukan oleh seseorang ketika gangguan terdeteksi. Langkah keempat adalah pemrograman menggunakan visual studio. Metode yang digunakan dalam penelitian ini adalah *forward chaining*.

1. Analisis Data

Tahap awal ini dilakukan analisa kebutuhan, analisa permasalahan yang muncul, analisa *user*. Metode yang biasa digunakan pada tahap ini adalah membaca manual dokumentasi, pada analisis awal ini juga dilakukan dengan mencari informasi yang pernah dibuat sebelumnya.
Metode data yang dipakai pada penelitian ini adalah deskritif kualitatif. Sedangkan metode analisis berorientasi objek yang digunakan pada penelitian ini adalah metode *Unified* (Hariyanto, 2004). Adapun tahapan analisis yang digunakan adalah :

a. Berpedoman pada kebutuhan pemakai sistem
b. Mengidentifikasi skenario pemakaian atau *usecase*
c. Memilih kelas-kelas dan objek menggunakan kebutuhan penuntun.
d. Mengidentifikasi atribut dan operasi masing-masing kelas objek
e. Mengidentifikasi struktur dan hirarki kelas-kelas.
f. Membangun model keterhubungan kelas dan objek

2. Desain

Desain antarmuka dalam penelitian ini berupa halaman *login*, desain halaman berhasil *login*, desain menu utama dan desain jika terjadi serangan yang akan memberikan gambaran jelas tentang penelitian. Biasanya hasil dari desain berupa.

a. *Use case*, kelas diagram dan tampilan visual sistem pakar
b. Gambar-gambar detil estimasi kebutuhan yang ada

3. Implementasi

Dalam fase implementasi akan diterapkan semua yang telah direncanakan dan pembuatan modul yang telah dirancang sebelumnya sesuai dengan bahasa pemograman yang digunakan dalam sistem yang akan dibangun. Implementasi sistem akan dilakukan dengan spesifikasi berikut:

a. Sistem operasi *Windows 7*
b. Memori 2 GB
c. Bahasa Pemrograman *C++*
d. *Compiler* Visual Studio

2. HASIL DAN PEMBAHASAN

Penelitian ini hanya didasarkan pada simulasi aplikasi yang bersifat *offline* tanpa dilakukan pengujian. Pengujian pengguna akan dilakukan di riset pengembangan selanjutnya ketika aplikasi benar-benar dijalankan dalam jaringan internet. Pada gambar 1 bisa dilihat dengan jelas bahwa sistem pakar memiliki *rule base*, *database* dan *inference engine*. Mesin inferensi yang dikembangkan dalam sistem pakar ini menggunakan aturan untuk menghasilkan hasil diagnosis berdasarkan data dan fakta penelitian sebelumnya dan melakukan pengukuran dengan *Intrusion Index* yang menggunakan perhitungan sebagai berikut (Sodiya et al., 2007).

R1 : *IF II is low THEN Deflect*
R2 : *IF II is high THEN Prevent*
R3 *: IF II is very high THEN Preempt*

$$\textit{Intrusion Index}\ (\mathrm{II}) = \frac{\sum_{i=1}^{n} X_i}{\square}$$
$$\sum_{i=1}^{n} X_{i\ \max()}$$

Dimana: untuk *n* = 3 (karena ada tiga variabel)

Variabel untuk perhitungan tersebut adalah

a)	Kategori Serangan	Skor
	Confidentiality	1
	Integrity	2
	Availability	3
b)	Dampak Serangan	Skor
	Low	1
	High	2
	Very High	3
c)	Tingkat Pelanggaran Keamanan	
	Low	1
	High	2
	Very High	3

X = Variabel skor
$X_{i\ \max(\ )}$ = Skor maksimum yang diperoleh untuk variabel *i*
Nilai maksimum untuk *instrusion* adalah 1 (satu), yang mana dibagi menjadi tiga rating sebagai berikut:

1. *Intrusion Index* akan bernilai rendah ketika $0 \leq II < 0{,}3$
2. *Intrusion Index* akan bernilai tinggi ketika $0{,}3 \leq II < 0{,}7$
3. *Intrusion Index* akan bernilai sangat tinggi ketika $0{,}7 \leq II < 1$

Ini berarti bahwa jika indeks yang dihitung antara 0 dan 0,3, maka Intrusin Indek bernilai rendah, dan seterusnya.

Jika terjadi sebuah serangan dengan kategori *confidentiality* yang bernilai 1, *attack implication* bernilai 3, dan *security violation level* bernilai 2. maka perhitungan nya sebagai berikut:

$$\sum_{i=1}^{3} 1+3+2=6$$

dan untuk nilai $Xi_{max(\ ),}$ nilai yang di dapat terdiri dari skor tertinggi yaitu *attack category* bernilai 3, *attack implication* bernilai 3, dan *security violation level* bernilai 3:

$$\sum_{i=1}^{3} 3+3+3=9$$

sehingga :

$$= \frac{\sum_{i=1}^{n} Xi}{\sum_{i=1}^{n} Xi \max()}$$

$$= \frac{\sum_{i=1}^{3} 1+3+2=6}{\sum_{i=1}^{3} 3+3+3=9}$$

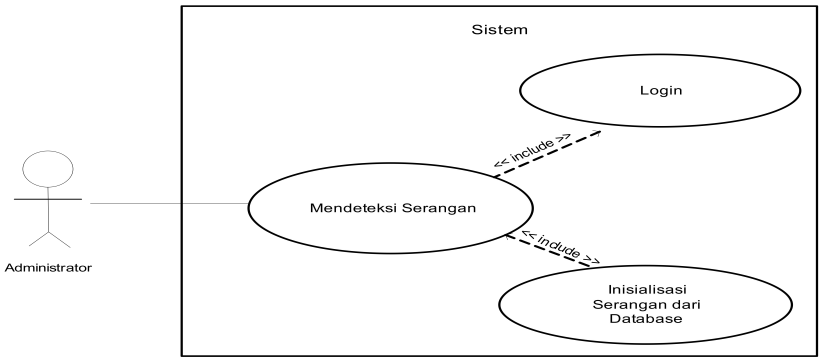
$$= \frac{6}{9} \text{ atau } \frac{2}{3}$$

$$= 0{,}6\ (\text{High})$$

Dalam simulasi ini *rule base* di klasifikasikan menjadi tiga macam pesan yaitu:

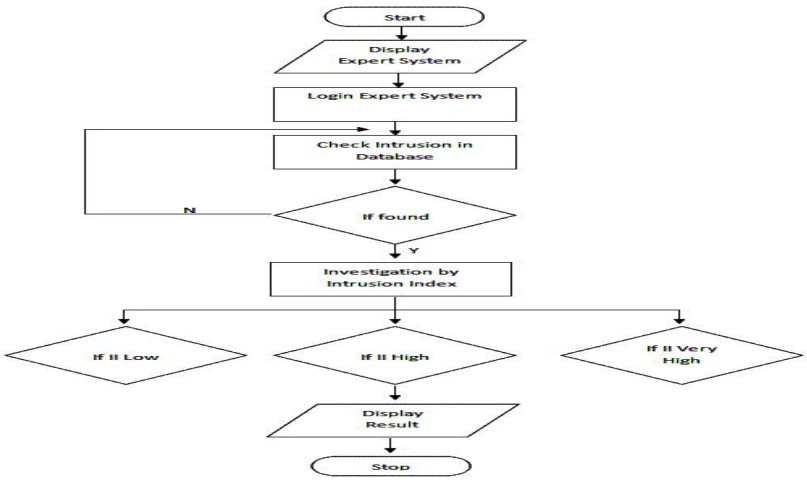
1. *Deflect* : Memberikan peringatan ke admin agar waspada terhadap serangan.
2. *Prevent*: Memberikan peringatan ke admin agar mencegah penyusup dari jaringan lokal.
3. *Preempt*: Memberikan peringatan ke admin agar memutuskan koneksi komputer atau PC yang telah diserang

Pada gambar 2 diagram *use case* bisa dilihat bahwa aktor utama pada sistem ini adalah seorang administrator yang sudah terverifikasi dan harus *login* terlebih dahulu sebelum menggunakan sistem. Administrator otomatis menuju halaman deteksi serangan untuk dapat melihat atau memonitoring kondisi jaringan. *Use case* diagram merupakan suatu model yang fungsinal dalam sistem yang menggunakan aktor dan use *case* (Aditiawarman, 2017).



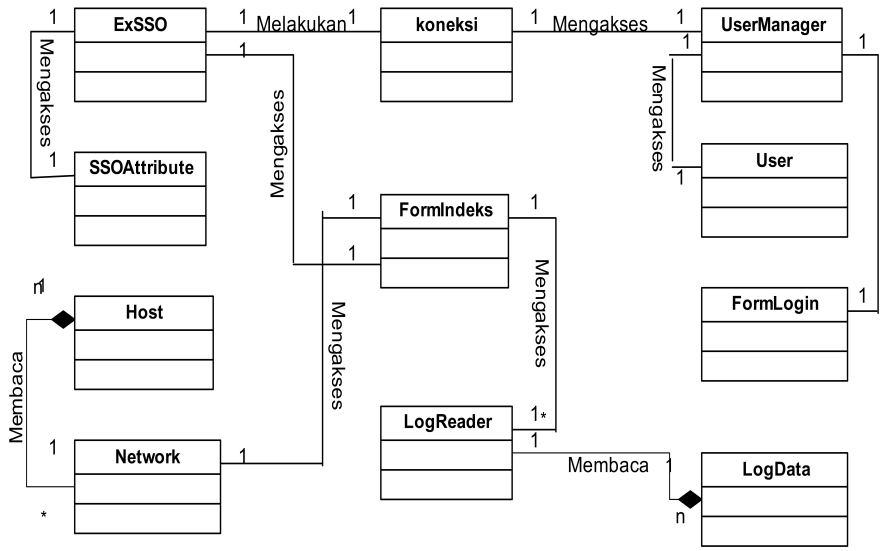
Gambar 1 Diagram *Use Case*

Gambar 1 menjelaskan bagaimana sistem pakar bekerja ketika seorang administrator telah berhasil login maka sistem akan melakukan pengecekan di database dan mengelola database tersebut dengan perhitungan *intrusion index* jika terjadi serangan apakah termasuk kategori *low, high* dan *very high* sehingga sistem pakar akan otomatis menampilkan hasil keputusan apa yang harus dilakukan oleh seorang administrator dalam menyikapi serangan yang terjadi.



Gambar 2 *Flow chart*

Pada gambar 4 terdapat 11 (sebelas) kelas yaitu *Exsso, SSOAttribute,Host, Network, Koneksi, FormIndeks, Logreader, user manager, user, Form Login* dan *Log Data*, dimana masing-masing kelas mempunyai *method* dan atribut.



Gambar 3 Kelas Diagram Keseluruhan

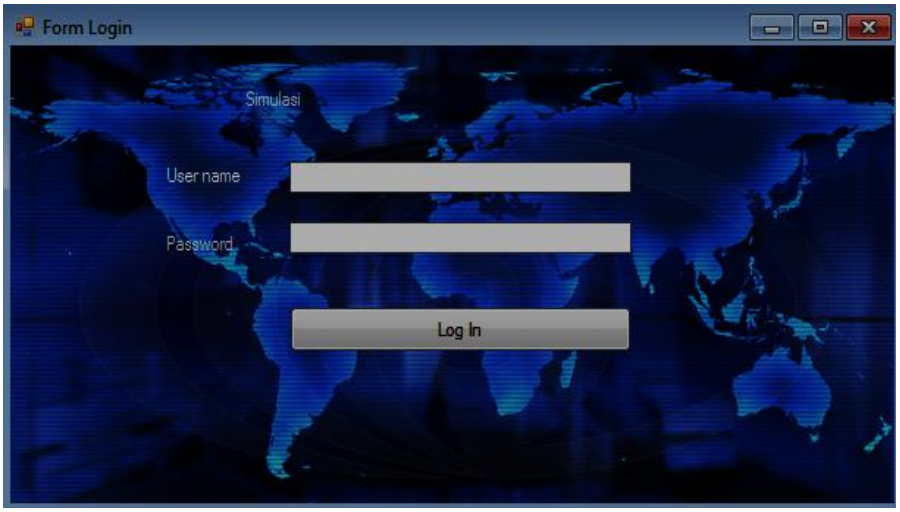
Kelas ExSSO merupakan kelas kontrol yang menangani perhitungan sistem pakar pada sistem. Kelas *SSOAttribute* merupakan kelas entitas yang menangani data-data ke database. Kelas *Host* merupakan kelas entitas yang berfungsi menyediakan data-data informasi komputer penyerang. Kelas *Network* merupakan kelas kontrol yang menghubungkan kelas *Host* dan kelas antarmuka *FormIndeks*. Kelas koneksi merupakan kelas koneksi yang menghubungkan

kelas ExSSO dan kelas *UserManager*. Kelas *FormIndeks* merupakan kelas antarmuka yang menangani tampilan menu utama dan berperan sebagai induk aplikasi. Kelas *LogReader* adalah kelas yang berfungsi membaca *filelog* untuk ditampilkan pada sistem. Kelas *UserManager* merupakan kelas *control* yang mengatur *username* dan *password* admin. Kelas *User* merupakan kelas entitas untuk *username* dan *password* admin. Kelas *FormLogin* merupakan kelas antarmuka yang menangani tampilan menu *Login* untuk mengakses sistem. Kelas *LogData* merupakan kelas yang mengatur data-data *log* seperti nama serangan,alamat IP,waktu,dan tanggal serangan.

Tabel 1 Seluruh Nilai *Intrusion Index* yang terpenuhi

No	*Security Violation Level*	*Attack Implication*	*Attack Category*	Hasil	Nilai *Intrusion Index*	Kategori Serangan
1.	1	1	1	3	0,3	Low
2.	1	1	2	4	0,4	High
3.	1	1	3	5	0,5	High
4.	1	2	1	4	0,4	High
5.	1	2	2	5	0,5	High
6.	1	2	3	6	0,6	High
7.	1	3	1	5	0,5	High
8.	1	3	2	6	0,6	High
9.	1	3	3	7	0,7	Very High
10.	2	1	1	4	0,4	High
11.	2	1	2	5	0,5	High
12.	2	1	3	6	0,6	High
13.	2	2	1	5	0,5	High
14.	2	2	2	6	0,6	High
15.	2	2	3	7	0,7	Very High
16.	2	3	1	6	0,6	High
17.	2	3	2	7	0,7	Very High
18.	2	3	3	8	0,8	Very High
19.	3	1	1	5	0,5	High
20.	3	1	2	6	0,6	High
21.	3	1	3	7	0,7	Very High
22.	3	2	1	6	0,6	High
23.	3	2	2	7	0,7	Very High
24.	3	2	3	8	0,8	Very High
25.	3	3	1	7	0,7	Very High
26.	3	3	2	8	0,8	Very High
27.	3	3	3	9	1	Very High

Rancangan antarmuka perangkat lunak yang dibangun. Sesuai analisa, ada beberapa kelas antarmuka (*interface*) yaitu *Form Login* dan *Form Indeks*. Adapun rancangan tampilan *Form* Utama sebagai berikut.



Gambar 4 Antarmuka *Form Login*

Pada gambar 8 Tampilan *form* apabila berhasil login dengan kecocokan *username* dan *password* yang diterima dan terdaftar di database. Saat tampilan berhasil login sudah ditampilkan maka form Indeks siap digunakan.

Gambar 5 Antarmuka berhasil *login*

Tampilan *form* apabila berhasil login dengan kecocokan *username* dan *password* yang diterima dan terdaftar di database. Saat tampilan berhasil login sudah ditampilkan maka form Indeks siap digunakan.

Gambar 6 Antarmuka *Form Indeks*

KESIMPULAN

Kesimpulan yang dapat ditarik berdasarkan hasil dari desain dan implementasi analisis basis data sistem pakar menggunakan metode *forward chaining* adalah berjalan dengan baik dan mengeluarkan hasil nilai sesuai yang diterapkan Sistem pakar adalah solusi yang nantiny akan membantu seorang administrator jaringan internet tentang keputusan apa yang diambil ketika terjadi serangan pada jaringan. Perhitungan *Intrusion index* sangat membantu dalam penentuan nilai yang berguna sebagai mesin inferensi sistem pakar. Sistem pakar ini diharapkan membantu akuisisi pengetahuan dari pakar ke sistem dan tidak menutup kemungkinan adanya pembaruan akuisisi pengetahuan dari pakar. Selain itu, memperbarui aturan juga menjamin yang baru pengetahuan dari para ahli karena tidak harus merusak aturan dasar sistem. Tampilan antarmuka yang berkonsep visual grafis dapat memudahkan seorang administrator membaca informasi yang dikeluarkan oleh sistem pakar.

Keterbatasan penelitian adalah sistemnya masih dirancang hanya diterapkan sebagai simulasi dan sistem pakar ini belum diterapkan secara *online*. Karena itu, disarankan untuk penelitian masa depan atau selanjutnya untuk mengembangkan sistem pakar ini dengan bersifat *real time.*

DAFTAR PUSTAKA

Aditiawarman. (2017). *Sistem Pakar Pendeteksi Penyakit Mata Berbasis Android*. *5*(2).

Desiani, A. (2006). *Konsep Kecerdasan Buatan*. Yogyakarta: Andi Offset.

Hariyanto, B. (2004). *Sistem Manajemen Basis Data*. Bandug: Informatika.

Sodiya, A. S., Adeniran, O., & Ikuomola, R. (2007). An Expert System-based Site Security

Officer. *Journal of Computing and Information Technology*, *15*(3), 227–235.

https://doi.org/10.2498/cit.1000961

Turban, efraim. (1995). *Decision support and expert systems : management support systems*.

JIFSKa, Vol. 4, No. 3, JANUARI, 2020, Pp. 173 – 184
ISSN : 2527 – 5836 (print) | 2528 – 0074 (online)

Pengembangan Sistem Pemetaan Status Mutu Air Sungai Berbasis Web Menggunakan *Extreme Programming*

Shofwatul 'Uyun[(1)], Ramadhan Salahudin Al Ayubi[(2)], Yulia Siti Ambarwati[(3)]

Teknik Informatika Universitas Islam Negeri Sunan Kalijaga Yogyakarta
Jl. Marsda Adisucipto No 1 Yogyakarta 55281
e-mail : shofwatul.uyun@uin-suka.ac.id, 16650065@student.uin-suka.ac.id, 16650078@student.uin-suka.ac.id

Abstract

The high water pollution index causes a decrease in water quality so that it can interfere with the health of living things. In order to overcome this, the government has tried to monitor water quality whose results can be known by the community. However, information disclosure and ease of accessing information are felt to be lacking. This study aims to present information about the quality status of river water and its relatively up-to-date and easily accessed by the public online. The storet method is used to determine the status of river water quality with seven parameters: temperature, EC, TDS, pH, DO, BOD and E.coli. The features provided will be explained in the results and discussion presented in several UML diagrams. In order to get results that match user expectations, this system was developed with extreme programming system development methods. The results of testing the functionality of the system to users, volunteers, and admins were found to be 100%, 99.26%, and 98.96%. While system reusability testing received 62.65% of responses strongly agreed, 36.80% agreed, 0.55% disagreed and 0% strongly disagreed.

Keywords: Pollution, River Water, Extreme Programming

Abstrak

Tingginya indeks pencemaran air mengakibatkan menurunnya kualitas air sehingga dapat mengganggu kesehatan mahluk hidup. Dalam rangka menanggulangi hal tersebut, pemerintah telah berupaya melakukan pemantauan kualitas air yang hasilnya dapat diketahui oleh masyarakat. Namun, keterbukaan informasi dan kemudahan dalam pengaksesan informasi dirasa masih kurang. Penelitian ini bertujuan untuk menyajikan informasi mengenai status kualitas mutu air sungai secara daring dan relatif uptodate serta mudah diakses oleh masyarakat secara online. Metode storet digunakan untuk menentuan status mutu air sungai dengan tujuh parameter yaitu : temperatur, EC, TDS, pH, DO, BOD dan E.coli. Fitur-fitur yang disediakan akan dijelaskan pada bagian hasil dan pembahasan yang disajikan dalam beberapa diagram UML. Dalam rangka mendapatkan hasil yang sesuai harapan pengguna, sistem ini dikembangakan dengan metode pengembangan sistem extreme programming. Adapun hasil pengujian fungsionalitas sistem kepada user, relawan, dan admin didapatkan hasil sebesar 100%, 99,26%, dan 98,96%. Sedangkan pengujian usabilitas sistem mendapatkan respon sangat setuju sebanyak 62,65%, setuju 36,80%, tidak setuju 0,55% dan sangat tidak setuju 0%.

Kata Kunci : Pencemaran, Air Sungai, Sistem, Online, Extreme Programming

1. PENDAHULUAN

Indonesia merupakan negara yang 3/4 dari wilayahnya merupakan perairan (Statistik, 2016). Dari 3/4 wilayah perairan, Indonesia memiliki lebih dari 5.590 sungai (Samekto & Winata, 2016) yang tersebar di berbagai wilayah. Berdasarkan data yang dirilis oleh Badan Pusat Statistik (2017) mengenai Status Kualitas Air Sungai, sebanyak 59% sungai mengalami cemar berat pada tahun 2016. Akan tetapi informasi yang disajikan dalam Statistik Lingkungan Hidup Indonesia 2017 masih terbatas pada waktu yang kurang uptodate dan keterbatasan informasi mengenai data penelitian kualitas air sungai. Data yang ditampilkan hanya berupa status kualitas air sungai apakah tercemar ringan, berat, atau sedang. Tidak dijelaskan lebih detail

mengenai hasil pengujian sehingga data yang disajikan kurang memberi pemahaman masyarakat akan kualitas air sungai.

Tingginya pemanfaatan air sungai sebagaimana dikemukakan oleh Samekto & Winata (2016) sekitar 32 milyar meter kubik air per tahun dimanfaatkan untuk memenuhi kebutuhan rumah tangga, kota, dan industri, sedangkan untuk kebutuhan irigasi dibutuhkan sekitar 128 milyar meter kubik, menjadikan informasi mengenai kualitas status mutu air sungai sangat penting bagi masyarakat sekitar. Konsumsi air nasional tertinggi digunakan untuk memenuhi kebutuhan irigasi yang diperlukan untuk mengairi persawahan guna memenuhi target kebutuhan konsumsi pangan (PAwitan et al., 2011).

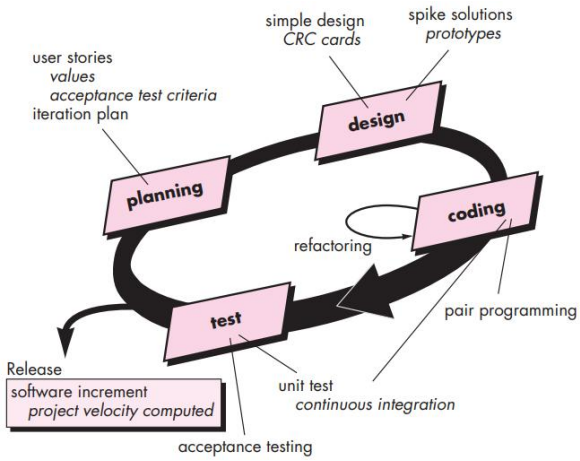
Pentingnya sungai bagi kelangsungan hidup yang telah dipaparkan diatas telah meningkatkan kesadaran masyarakat untuk peduli terhadap kebersihan sungai. Dalam hal ini peneliti mengusulkan pengembangan sistem untuk pemantauan status mutu air sungai dengan melibatkan masayarakat di sekitar daerah aliran sungai, khususnya para komunitas pecinta sungai atau lingkungan. Beberapa komunitas pegiat lingkungan yang telah bekerja sama dalam penelitian ini adalah Forum Komunikasi Daerah Aliran Sungai (Forsidas) Gajah Wong di Yogyakarta dan komunitas pegiat Sungai Brantas di Jombang. Akan tetapi mereka belum memiliki media untuk menyalurkan hasil pengamatan agar dapat diketahui oleh masyarakat luas. Oleh karenanya diperlukan sebuah wadah untuk menampung hasil pengujian yang telah dilakukan oleh para komunitas pecinta sungai tersebut. Kemudian hasilnya dapat diakses oleh masyarakat luas. Keterbatasan KLHK dalam menyampaikan hasil pengujian terhadap kualitas air sungai secara daring dan real time mendorong untuk membuat alternatif yang menjembatani peneliti dan masyarakat agar dapat berkomunikasi dengan lebih baik. Dalam hal ini adalah mengkomunikasikan data hasil penelitian terhadap kualitas mutu air sungai dalam bentuk aplikasi berbasis web yang dapat diakses secara online dan real time. Selain itu, para relawan yang tergabung dalam komunitas pecinta sungai dapat ikut berperan aktif untuk memasukkan data penelitian kualitas air sungai dalam aplikasi ini yang sebelumnya telah diberikan bimbingan teknis mengenai pengujian air sungai dengan melakukan pengukuran dan penggunaan aplikasi.

Ada tujuh parameter air sungai yang digunakan sebagai dasar penentuan status mutu air sungai termasuk dalam empat kategori: baik, tercemar ringan, tercemar sedang den tercemar berat, antara lain : secara fisika (temperatur, EC, TDS), kimia (pH) dan biologi (DO, BOD dan E.coli). Parameter ini dipilih karena dirasa cukup mewakili penilaian secara fisika, kimia dan biologi. Metode Storet digunakan untuk menentukan status mutu air pada Sistem Pemetaan Status Mutu Air Sungai Berbasis Web. Metode Storet menentukan status mutu air dengan membandingkan nilai parameter hasil pengujian dan standar baku mutu yang telah ditetapkan oleh peraturan yang berlaku (Hidup, 2003). Anwar, Hariono, Wibowo, & Dyah Utami (2018) menggunakan metode Storet untuk mengukur sifat fisik, kimia, dan mikrobiologis dalam menentukan status mutu air sungai. Selain metode Storet, terdapat metode IP(Indeks Pencemaran), CCME WQI(Canadian Council of Ministers of The Environment) untuk menghitung kualitas mutu air sungai (Romdania, Herison, Susilo, & Novilyansa, 2018). Pedoman yang digunakan untuk menentukan status mutu air dengan Metode Storet dilakukan sesuai dengan peraturan yang berlaku di daerah tersebut. Sebagai contoh, untuk menentukan status mutu air di Sungai Gajah Wong, pedoman baku mutu yang digunakan adalah PP DIY No 20 tahun 2008.

Setelah melakukan penelitian terhadap kualitas mutu air sungai, para penggiat sungai yang bertindak sebagai relawan memerlukan wadah untuk menampung hasil pengujiannya. Sistem ini menjadi alternatif bagi relawan yang ingin menyampaikann hasil pengujiannya kepada masyarakat. Selanjutnya sistem akan memroses data hasil pengujian dan akan menampilkan status mutu air sungai beserta data hasil pengujian dengan data yang relative uptodate. Sistem ini dikembangkan menggunakan metode pengembangan sistem Extreme Programming. Extreme programming dipilih karena metode ini sesuai apabila dihadapkan dengan perubahan requirement dengan cepat (Supriyatna, 2018). Metode ini menjadikan spesifikasi perangkat lunak menjadi bagian yang paling penting dari pengembangan sistem, tidak hanya bergantung kepada kecerdasan tim pengembang (Yadav, Yasvi, & Shubhika, 2019). Pada awal tahap pengembangan sistem tidak sedikit perubahan terjadi mengikuti requirement dari pengguna, baik dari masyarakat maupun pihak pakar. Oleh karena itu metode extreme programming dipilih untuk mengembangkan Sistem Pemetaan Status Mutu Air Sungai Berbasis Web.

2. METODE PENELITIAN

Metodologi yang digunakan dalam pengembangan Sistem Pemetaan Status Mutu Air Sungai Berbasis Web adalah metode Extreme programming. Extreme Programming merupakan metode pengembangan software yang berfokus pada peningkatan kualitas software dan perubahan kebutuhan customer. Metode ini menggabungkan feedback dari customer dan menggunakan pendekatan kerja tim yang menjadikannya fleksibel dan efektif untuk digunakan dalam pendekatan pengembanga software. Menurut Pressman (2010) terdapat empat tahap pengembangan dalam siklus Extreme Programming, antara lain : Planning, Design, Coding, dan Testing yang ditunjukkann pada Gambar 2.1.



Gambar 2.1 Tahapan Extreme Programming menurut Pressman (2010)

2.1. Planning

Pada tahapan planning atau yang bisa disebut dengan perencanaan merupakan tahapan mengumpulkan permintaan sistem dan analisis fisibilitas (user stories) dan menentukan kelompok stories yang akan dikembangkan pada rilis berikutnya (Pressman, 2010). Proses ini dimulai dengan menentukan kemampuan apa saja yang dapat dilakukan oleh sistem, diantanya adalah melakukan pencarian terhadap sungai berdasarkan pulau, mengetahui bagaimana kualitas status mutu air sungai tersebut.

2.2. Design

Tahapan design melakukan perancangan sistem berdasarkan pada user stories yang telah dibuat dengan menekankan pada konsep kesederhanaan. Tahap ini merekomendasikan penggunaan kartu CRC (Class-Responsibility-Collaborator) dan solusi spike (Pressman, 2010) untuk mengimplementasikan desain prototype sistem. Desain prototype sistem dibuat dengan membuat UML (Unified Modeling Language), merupakan Bahasa permodelan perangkat lunak yang berorientasi objek sehingga mampu menjelaskan sistem secara detail (Suendri, 2018). Adapun permodelan UML yang digunakan antara lain : diagram use case, diagram activity, diagram sequence, dan diagram state.

2.3. Coding

Tahap coding merupakan implementasi desain pengembangan sistem kedalam bentuk user interface menggunakan bahasa pemrograman (Carolina & Supriyatna, 2019) dengan standar yang telah disepakati (Pambudi, 2016). Hal ini bertujuan agar mudah dibaca dan dipahami serta disatukan apabila dikerjakan lebih dari satu orang. Pengkodean dimulai dengan membuat desain antar muka aplikasi dilanjut membuat backend program untuk melakukan perhitungan terhadap status mutu air sungai.

2.4. Testing

Pada tahap testing atau pengujian program berfokus pada keseluruhan fitur dan fungsional sistem yang dapat ditinjau oleh customer. Tahap ini juga berfungsi untuk mendeteksi bug dan

menjadikan keberhasilan penggunaan oleh user sebagai indikator keberhasilan pengujian (Yadav et al., 2019). Pengujian dilakukan oleh pihak relawan / masyarakat yang menginputkan hasil pengujian air sungai ke dalam sistem. Hasil perhitungan oleh system kemudian dibandingkan dengan perhitungan manual oleh pakar biologi.

3. HASIL DAN PEMBAHASAN

3.1. Planning Tahap 1

Planning atau perencanaan merupakan tahapan mengumpulkan permintaan sistem dan analisis fisibilitas (user stories) dan menentukan kelompok stories yang akan dikembangkan pada rilis berikutnya. Tahap ini bertujuan untuk dapat mengidentifikasi atau mengevaluasi berbagai macam masalah maupun hambatan yang akan timbul pada sistem sehingga nantinya dapat dilakukan penanggulangan, perbaikan atau juga pengembangan. Tahap perencanaan dimulai dengan mendefinisikan actor apa saja yang terlibat dengan sistem, adalah sebagai berikut :

a. Admin

Admin merupakan pengguna yang memiliki hak akses paling tinggi diantara pengguna lainnya diantaranya meliputi pembuatan peta sungai, memverifikasi data pengamatan sungai, dan melakukan CRUD (Create, Read, Update, Delete) terhadap data relawan.

b. Relawan

Relawan terdiri dari beberapa komunitas sungai yang melakukan pengambilan sampel air di sungai kemudian melakukan input data hasil pengamatan ke sistem untuk selanjutnya akan diverifikasi oleh Admin.

c. User

Adalah pengguna yang dapat mengakses sistem secara online untuk mengetahui kualitas air sungai yang disediakan sistem setelah melalui proses pengambilan, perhitungan oleh sistem, dan verifikasi Admin.

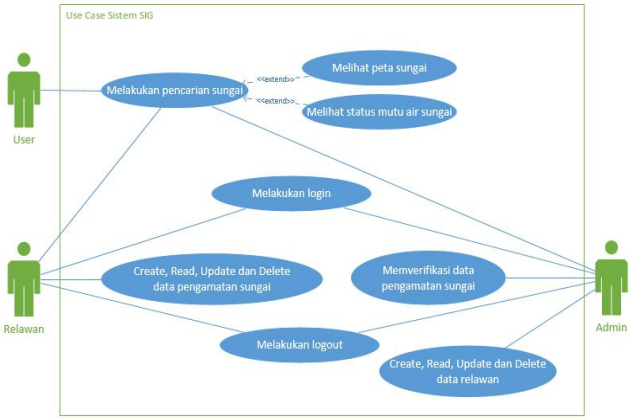
Selanjutnya menentukan kemampuan apa saja yang dapat dilakukan oleh sistem (membuat user stories). Selain menentukan user stories, pada tahap ini juga dilakukan konsultasi sistem dengan pihak pakar biologi.

3.2. Design Tahap 1

Design / perancangan sistem bertujuan untuk memberikan gambaran mengenai system berdasarkan pada user stories yang telah dibuat. Perancangan sistem merupakan tahap mendesain dari hasil proses perencanaan. Aspek fungsionalitas sistem digambarkan dengan menggunakan UML (Unified Modelling Language) yang terdiri dari diagram use case, diagram activity, diagram sequence, dan diagram state.

3.2.1. Use Case Diagram

Use case diagram yang digunakan dalam penelitian ini ditunjukkan oleh Gambar 3.1.



Gambar 3.1 Use Case Diagram

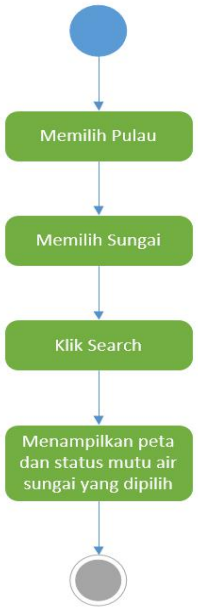
Dalam sistem pemantauan status mutu air sungai yang dikembangkan ada 3 aktor yang terlibat, yaitu User, Relawan dan Admin. Ketiga aktor tersebut memiliki hak dan kewajiban yang berbeda-beda sesuai dengan fungsinya. Aktor User dapat melakukan pencarian sungai yang akan menampilkan peta sungai dan status mutu air dari sungai yang dicari berdasarkan kriteria pulau, misal pulau jawa, sumatra, kalimantan, nusa tenggara timur dan lannya. Sedangkan aktor Relawan dapat melakukan login, pencarian sungai, create, read, update, dan delete data pengamatan sungai serta melakukan logout. Untuk aktor Admin dapat melakukan login, pencarian sungai, memverifikasi data pengamatan sungai, create, read, update, dan delete data relawan serta logout. Detail gambaran dari use case sistem pemantauan status mutu air sungai ditunjukkan pada Gambar 3.1

3.2.2 Activity Diagram

Setelah menemukan perilaku apa saja yang dilakukan oleh pengguna (user), yaitu admin maupun member di dalam sistem dengan menggunakan Diagram Use Case, tahap selanjutnya yaitu mengubah setiap aktivitas pengguna kedalam suatu Diagram Aktivitas (Activity Diagram), dan pada tahap ini akan didapatkan hasil alur yang terjadi ketika aktivitas tersebut berjalan.
Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, kemungkinan yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram dibuat berdasarkan sebuah atau beberapa use case pada use case diagram.

3.2.2.1 Activity Diagram User

Activity diagram User dimulai dengan : User memilih pulau, kemudian akan sistem akan menampilkan daftar sungai yang terdapat pada pulau tersebut. Setelah itu User memilih sungai yang akan dicari, selanjutnya klik button Search. Sistem akan memulai proses pencarian yang akan menampilkan peta dan status mutu air berdasarkan sungai yang dipilih, proses pun selesai. Detail gambaran dari use case sistem pemantauan status mutu air sungai ditunjukkan pada Gambar 3.2.

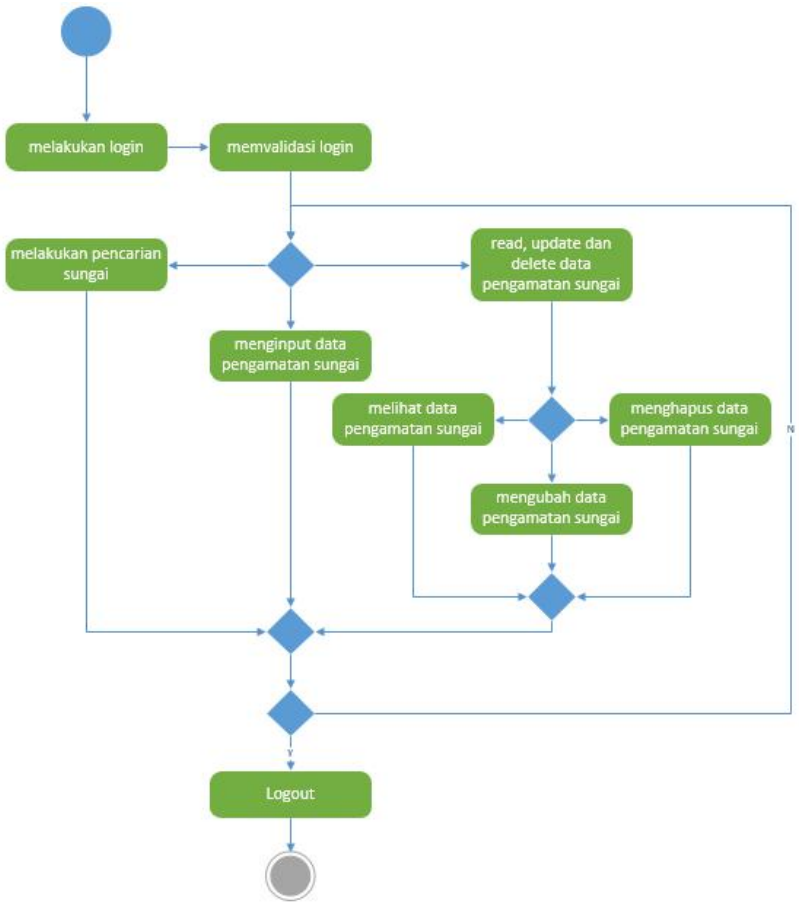


Gambar 3.2 Activity Diagram User

3.2.2.2 Activity Diagram

Activity diagram Relawan dimulai dengan : Relawan melakukan login, kemudian akan masuk ke proses memvalidasi login. Setelah login berhasil, Relawan dapat memilih menu yang dinginkan, antara lain: melakukan pencarian sungai, input data ataupun melakukan read, update dan delete data pengamatan sungai yang telah diinputkan , yang didalamnya meliputi fitur melihat data pengamatan sungai, mengubah data pengamatan sungai dan menghapus data pengamatan sungai. Selanjutnya Relawan dapat kembali memilih menu atau jika sudah tidak ada keperluan Relawan dapat melakukan logout, proses pun selesai. Detail gambaran dari

activity diagram relawan sistem pemantauan status mutu air sungai ditunjukkan pada Gambar 3.3.



Gambar 3.3 Activity Diagram

3.2.2.3 Activity Diagram Admin

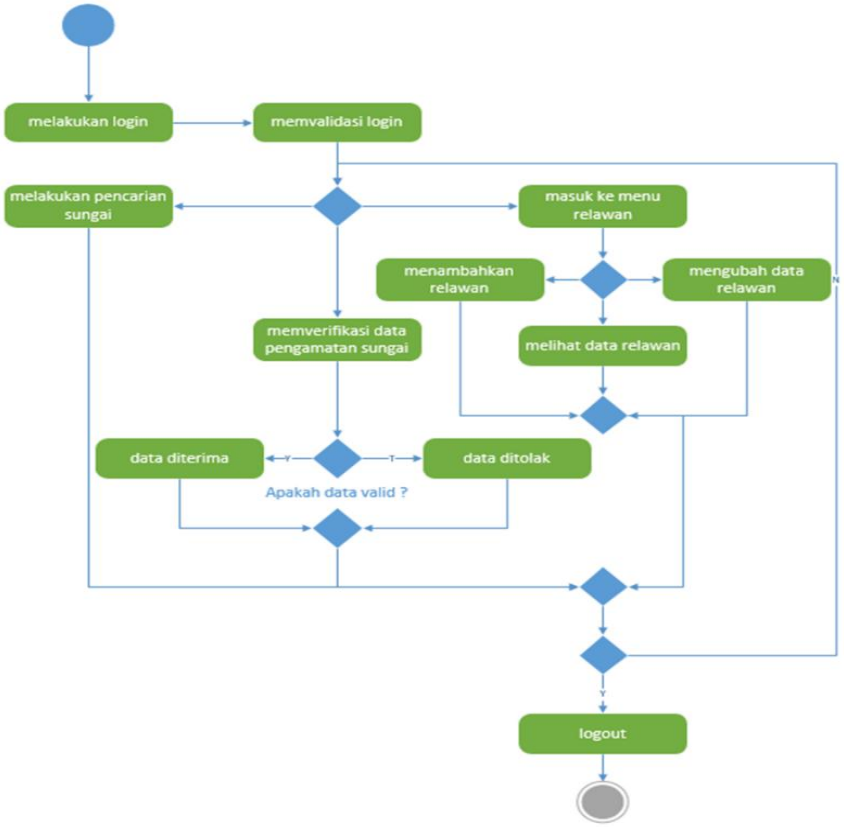
Activity diagram Admin dimulai dengan : Admin melakukan login, kemudian akan masuk ke proses memvalidasi login, selanjutnya Admin dapat memilih menu yang dinginkan, antara lain : melakukan pencarian sungai, memverifikasi data pengamatan sungai yang apabila data valid maka akan diterima jika tidak maka akan ditolak. Selain itu Admin dapat masuk ke menu relawan, yang didalamnya meliputi fitur menambahkan Relawan, melihat data Relawan, dan mengubah data relawan. Selanjutnya Admin dapat kembali memilih menu atau melakukan logout, proses pun selesai. Detail gambaran dari activity diagram admin sistem pemantauan status mutu air sungai ditunjukkan pada Gambar 3.4.

3.2.3 Sequence Diagram

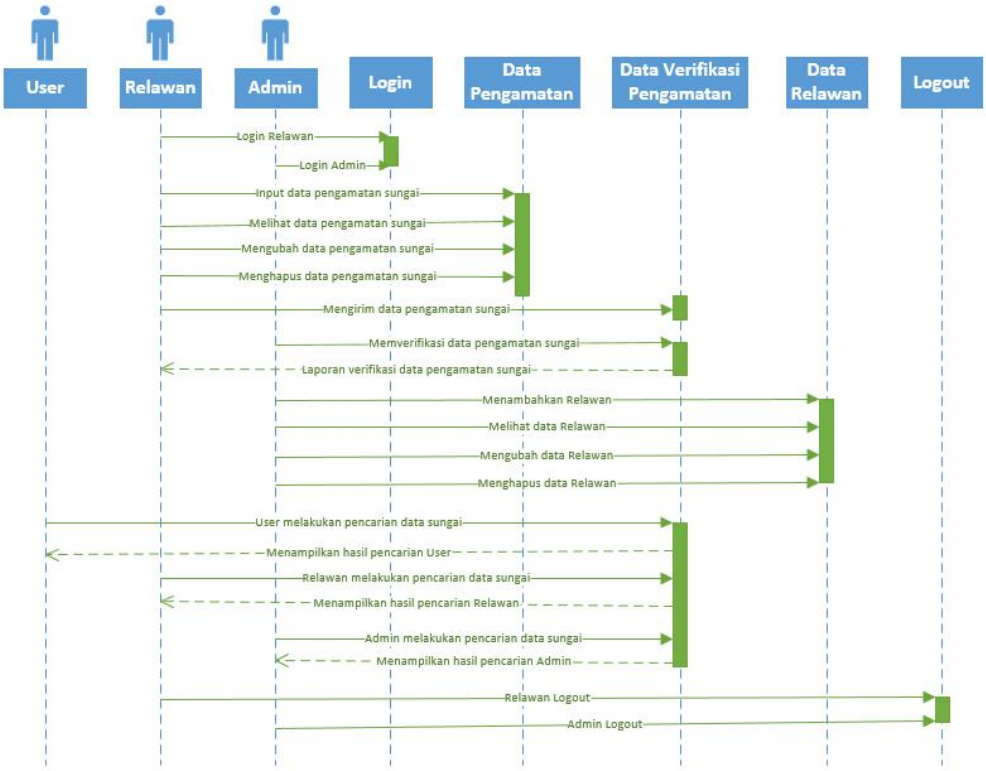
Sequence diagram pada sistem pemantauan status mutu air sungai memiliki alur proses pada sistem sebagai berikut :

a. Pada level User, yang dapat dilakukan yakni melakukan pencarian data pengamatan sungai yang akan menampilkan hasil pencarian berupa peta dan status mutu air.
b. Relawan melakukan login pada sistem dengan memasukkan username dan password. Pada level Relawan, yang dapat dilakukan yakni input data pengamatan sungai, melihat data pengamatan sungai, mengubah data pengamatan sungai, menghapus data pengamatan sungai, mengirimkan data pengamatan sungai yang akan mendapatkan feedback berupa laporan setelah melalui proses verifikasi oleh Admin, melakukan pencarian data pengamatan sungai dan melakukan logout.
c. Admin melakukan login pada sistem dengan memasukkan username dan password. Pada level Admin, yang dapat dilakukan yakni memverifikasi data pengamatan sungai, menambahkan relawan, melihat data relawan, mengubah data relawan, menghapus data relawan, melakukan pencarian data sungai, dan melakukan logout.

Detail gambaran dari sequence diagram sistem pemantauan status mutu air sungai ditunjukkan pada Gambar 3.5.



Gambar 3.4 Activity Diagram Admin



Gambar 3.5 Sequence Diagram

3.2.4 State Diagram

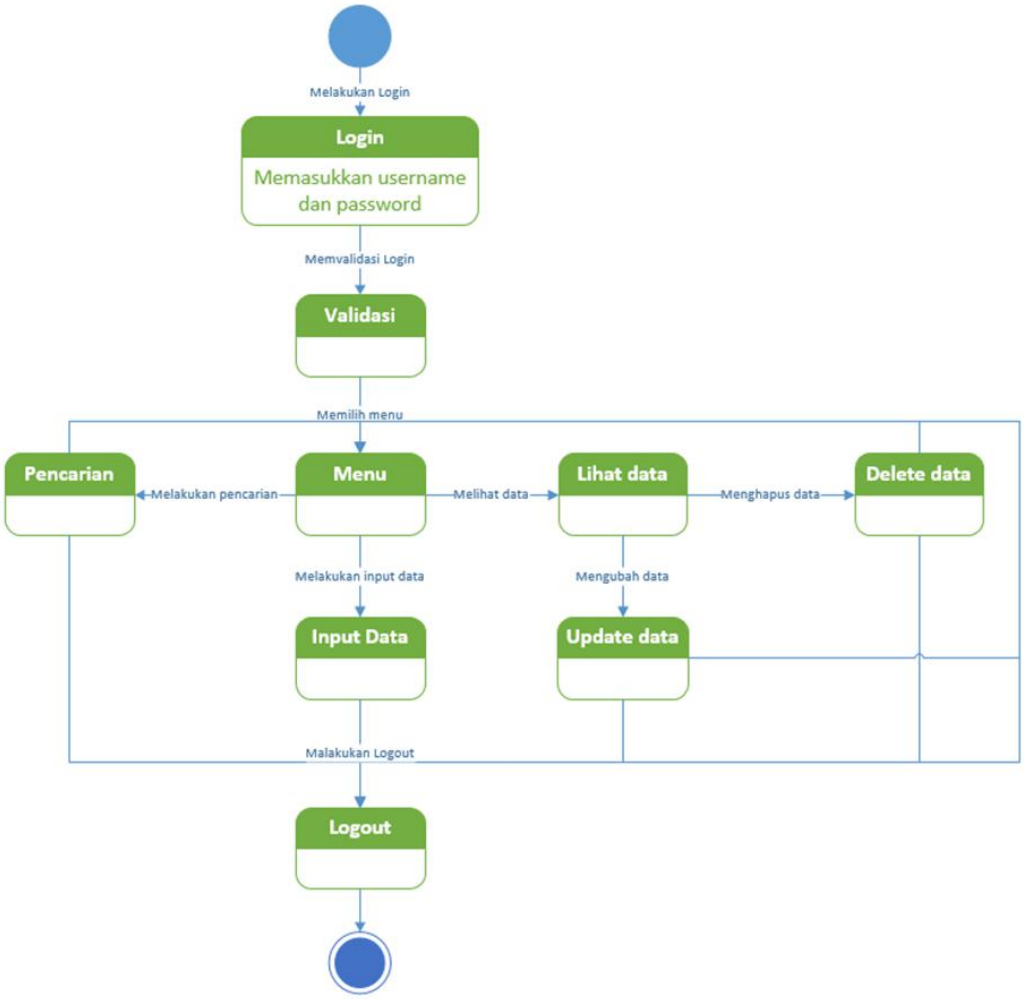
3.2.4.1 State Diagram User

Aliran data untuk state diagram user terdiri dari pemilihan pulau dilanjutkan dengan pemilihan nama sungai. Sistem akan otomatis menampilkan daftar sungai yang terdapat pada pulau yang telah dipilih. Setelah itu sistem akan melakukan proses searching berdasarkan kata kunci yang dimasukkan yaitu nama sungai dan pulau. Hasil dari pencarian sistem akan menampilkan map sungai serta status mutu air berdasarkan hasil kumulatif isian data untuk ketujuh paramater yang digunakan yang sebelumnya telah diinputkan oleh masing-masing relawan sungai dan telah dilakukan oleh validasi dari masing-masing admin sungai tersebut. Data yang ditunjukkan pada map sungai adalah data setiap titik koordinat lokasi pengambilan dan hasil kumulatif dari tiap koordinat dalam satu sungai dalam bentuk status mutu air sungai. User juga dapat melakukan klik pada titik koordinat yang diinginkan dan oleh sistem akan ditampilkan secara detail data kualitatif untuk setiap parameter. Detail gambaran dari state diagram user pada sistem pemantauan status mutu air sungai ditunjukkan pada Gambar 3.6.

Gambar 3.6 State Diagram User

3.2.4.2 State Diagram Relawan

Aliran data untuk state diagram relawan diawali dengan memasukkan user dan password, setelah relawan menginputkan data untuk login maka sistem akan memvalidasi apakah user dan password yang dimasukkan valid atau tidak. Setelah relawan sukses masuk ke sistem selanjutnya Relawan dapat menginputkan data hasil pengamatan sampel air sungai dengan tujuh parameter yang telah ditentukan untuk setiap titik koordinat. Pada setiap titik koordinat lokasi pengambilan, relawan dapat menginputkan data ulangan sebanyak tiga sampai lima kali ulangan, sesuai dengan berapa kali pengambilan sampel air pada suatu lokasi. Setelah menyimpan data pengamatan sampel air, relawan dapat melihat data, mengupdate data dan menghapus data yang dalam prakteknya didapatkan kesalahan input data dan ingin memperbaikinya. Data yang diinputkan oleh relawan tidak bisa secara otomatis ditampilkan oleh sistem, dikarenakan masih menunggu verifikasi dan moderasi dari admin masing-masing sungai. Relawan dapat melakukan logout apabila sudah selesai memasukkan data pengamatan dan sudah tidak memiliki keperluan dengan sistem. Detail gambaran dari state diagram user pada sistem pemantauan status mutu air sungai ditunjukkan pada Gambar 3.7.



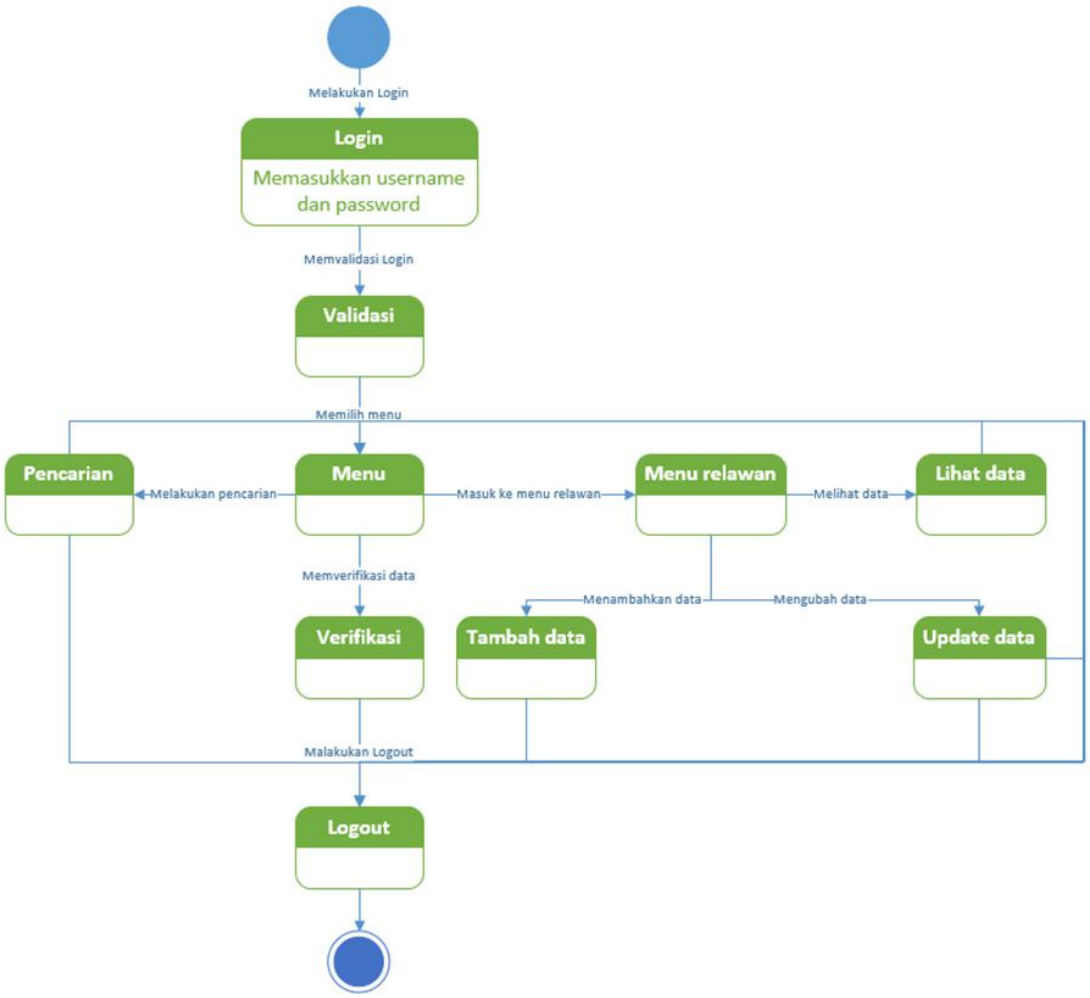
Gambar 3.7 State Diagram Relawan

3.2.4.3 State Diagram Admin

Aliran data untuk state diagram admin diawali dengan memasukkan user dan password, setelah admin menginputkan data untuk login maka sistem akan memvalidasi apakah user dan password yang dimasukkan sudah valid atau tidak. Setelah admin sukses masuk ke sistem selanjutkan dapat melakukan verifikasi dan validasi inputan data hasil pengamatan sampel air sungai untuk setiap titik koordinat oleh masing-masing relawan sungai. Setiap koordinat, admin dapat melakukan verifikasi inputan data ulangan sebanyak tiga sampai lima kali ulangan. Setelah data diverifikasi maka oleh sistem akan diakumulasikan dengan data-data dari koordinat lain untuk menampilkan status mutu air sungai secara otomatis dan real time. Detail gambaran dari state diagram admin pada sistem pemantauan status mutu air sungai ditunjukkan pada Gambar 3. 8.

3.3. Coding Tahap 1

Pada coding tahap 1 dilakukan implementasi desain UML kedalam bentuk user interface menggunakan bahasa pemrograman PHP dengan framework CodeIgniter. Pengkodean dimulai dengan membuat desain antar muka aplikasi, yang terdiri dari halaman user yang dapat diakses oleh semua pengguna tanpa harus melakukan login. Selanjutnya membuat halaman admin dan halaman relawan yang memerlukan hak akses untuk dapat masuk ke sistem ini. Setelah menyelesaikan user interface, lanjut ke pengkodean backend program dengan melakukan pemrograman untuk memfungsikan relawan dan admin dalam sistem. Selanjutnya membuat kode untuk melakukan pemrosesan hasil pengujian air sungai berdasarkan parameter yang telah ditentukan untuk selanjutnya menyajikan dalam tampilan yang menarik dan mudah dipahami oleh user. Hasil perhitungan sistem ditampilkan pada map sungai dengan data setiap titik koordinat lokasi pengambilan dan hasil akhir status mutu air sungai.



Gambar 3.8 State Diagram Admin

3.4. Pengujian Tahap 1

Tahap pengujian program berfokus pada keseluruhan fitur dan fungsional sistem yang dapat ditinjau langsung oleh customer. Setelah sistem selesai dibuat dilakukan pengujian dengan memasukkan data hasil pengujian air sungai kedalam sistem. Hal ini dilakukan oleh relawan yang sebelumnya telah diberikan hak akses oleh admin untuk selanjutnya melakukan login. Pengujian dilakukan terhadap tiga sungai yang ada di Indonesia. Dua diantaranya berada di Pulau Jawa, yakni Sungai Brantas di Jawa Timur dan Sungai Gajah Wong di DIY. Sisanya merupakan Sungai Lambanapu di Nusa Tenggara Timur, Sumba. Tahap memasukkan data telah berhasil hingga dapat menampilkan status mutu air sungai dari data sungai yang telah diinputkan, baik hasil tiap koordinat maupun dalam bentuk hasil akhir perhitungan keseluruhan koordinat. Namun pada tahap ini masih ada perbedaan terhadap perhitungan sistem dan perhitungan manual oleh pakar biologi. Setelah dilakukan penulusuran lebih lanjut, ditemukan adanya kesalahan terhadap baku mutu air yang dimasukkan, dalam hal ini kesalahan terdapat pada baku mutu air Sungai Lambanapu, Sumba.

3.5. Planning Tahap 2

Setelah mengetahui adanya kesalahan pada testing tahap 1, langkah selanjutnya adalah melakukan analisis terhadap kebutuhan yang akan digunakan dalam hal pengembangan sistem. Adapun data yang diperlukan adalah pedoman baku mutu berdasarkan peraturan yang berlaku pada daerah tersebut. Dalam hal ini digunakan pedoman baku mutu Lampiran Peraturan Pemerintah Nomor 82 Tahun 2001 (Pemerintah Republik Indonesia, 2001).

3.6. Design Tahap 2

Pada tahap ini tidak dilakukan perubahan, dikarenan kesalahan yang didapatkan pada tahap prngujian pertama tidak berprngaruh terhadap desain system. Melainkan kesalahan pada tahap coding.

3.7. Coding Tahap 2

Pada coding tahap 2 dilakukan perbaikan terhadap sistem pada bagian yang telah terindikasi terdapat error pada tahap pengujian pertama. Dalam hal ini memasukkan standar baku mutu yang sudah didapatkan pada tahap planning.

3.8. Pengujian Tahap 2

Pada tahap ini terlebih dahulu dilihat hasil pengujian terhadap kesalahan yang terjadi pada pengujian tahap pertama. Setelah hasil perhitungan sistem sudah sesuai dengan hasil perhitungan manual oleh pakar, maka lanjut pada pengujian sungai yang lain. Setelah program berjalan sesuai fungsinya, langkah selanjutnya melakukan pengujian berbasis kuesioner pada responden yang terdiri dari user biasa, relawan, dan admin. Adapun pengujiannya terdiri dari pengujian fungsional dan usabilitas sistem. Hasil pengujian yang didapatkan diperoleh dengan rumus Pers. (1).

$$\overline{X} = \frac{\sum X}{N} \times 100\% \qquad (1)$$

Keterangan :

$\overline{X}$: hasil rerata pengujian subyek X dalam persen

$\sum X$: Jumlah skor dalam distribusi subyek X

N : Banyaknya responden

Berdasarkan hasil pengujian yang disebar kepada 16 responden user biasa daidapatkan hasil 100 % user menyatakan bahwa fungsional sistem dapat berjalan dengan baik sesuai dengan yang diharapkan. Sedangkan hasil pengujian yang disebar kepada 16 responden relawan dapat diketahiu bahwa 99,26 % relawan menyatakan bahwa fungsional sistem dapat berjalan dengan baik sesuai dengan yang diharapkan. Sedangkan pengujian pada 16 responden admin , 98,96 % diantaranya menyatakan bahwa fungsional sistem dapat berjalan dengan baik sesuai dengan yang diharapkan. Pada pengujian usabilitas sistem Pemetaan Status Mutu Air Sungai Berbasis Web untuk user biasa yang melibatkan 16 responden, dapat diketahui bahwa sebagian besar pengguna menyatakan penilaian yang baik terhadap sistem pemetaan ini. Didapatkan hasil pengujian yang menunjukkan bahwa responden menyatakan sangat setuju sebanyak 64,58% , setuju 35,42 % , tidak setuju 0% , dan sangat tidak setuju 0%. Berdasarkan hasil pengujian dari segi usability sistem terhadap relawan yang melibatkan 16 responden, dapat diketahui bahwa sebagian besar responden menyatakan penilaian yang baik terhadap sistem pemetaan ini. Didapatkan hasil pengujian yang menunjukkann bahwa responden menyatakan sangat setuju sebanyak 62,98% , setuju sebanyak 37,02% , tidak setuju sebanyak 0% , dan sangat tidak setuju sebanyak 0%. Sedangkan hasil pengujian usabilitas system terhadap admin yang melibatkan 16 responden, dapat diketahui bahwa sebagian besar responden menyatakan penilaian yang baik terhadap sistem pemetaan ini. Didapatkan hasil pengujian yang menunjukkan bahwa responden menyatakan sangat setuju sebanyak 60,40% , setuju 37,95 % , tidak setuju 1,65% , dan sangat tidak setuju sebanyak 0%.

KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan terkait terhadap sistem Pemetaan Status Mutu Air Sungai dengan memanfaatkan teknologi informasi serta melibatkan komunitas penggiat sungai, maka dapat diambil kesimpulan bahwasanya sistem ini dapat membantu pemerintah maupun masyarakat dalam melakukan pemetaan status mutu air sungai di Indonesia dengan melibatkan peran aktif komunitas penggiat sungai dengan teknologi informasi. Hasil pengujian yang telah diolah oleh sistem dapat memberikan informasi mengenai status mutu air sungai berikut dengan hasil pengujiannya dengan akses yang mudah. Hal ini dapat dilihat dari hasil pengujian fungsionalitas sistem sebesar 100% terhadap responden user.

Dengan adanya sistem Pemetaan Status Mutu Air Sungai secara online, diharapkan mampu memberikan kemudahan bagi masyarakat untuk mengetahui status mutu air sungai yang ada di Indonesia sacara daring dan relatif up to date.

DAFTAR PUSTAKA

Anwar, S., Hariono, B., Wibowo, M. J., & Utami, M. M. D. (2018). Penentuan Status Mutu Air Metode Storet DAS Kali Curah Macan. *Jurnal Ilmiah Inovasi*, *18*(2), 95–98.

Carolina, I., & Supriyatna, A. (2019). Penereapan Metode Extreme Programming dalam Perancangan Aplikasi Perhitungan Kuota SKS Mengajar Dosen. *Jurnal IKRA-ITH Informatika*, *3*(1), 106–113.

Hidup, M. N. L. (2003). Keputusan Menteri Negara Lingkungan Hidup Nomor : 115 Tahun 2003 Tentang Pedoman Penentuan Status Mutu Air. *Jakarta : Menteri Negara Lingkungan Hidup*, pp. 1–15.

Pambudi, A. (2016). *Rancang Bangun Sistem Informasi Penilaian Kinerja Instruktur Training ICT Menggunakan Metode Extreme Programming*.

PAwitan, H., Adidarma, W., Hatmoko, W., Hadihardaja, I. K., Kodoatie, R. J., Putuhena, W. M., … Radhika. (2011). *Tapak Air dan Strategi Penyediaan Air di Indonesia*.

Pemerintah Daerah Istimewa Yogyakarta. (2008). *Lampiran 1 Peraturan Gubernur DIY No 20 Tahun 2008*.

Pemerintah Republik Indonesia. (2001). Peratuan Pemerintah Republik Indonesia Nomor 82 Tahun 2001. *Indonesia*.

Pressman, R. S. (2010). Software Software Engineering : A Practitioner's Approach, Seventh Edition. In *McGraw-Hill*.

Romdania, Y., Herison, A., Susilo, G. E., & Novilyansa, E. (2018). Kajian Penggunaan Metode IP, STORET, dan CCME WQI dalam Menentukan Status Kualitas Air. *Jurnal SPATIAL - Wahana Komunikasi Dan Informasi Geografi*, *18*(1), 1–13.

Samekto, C., & Winata, E. S. (2016). Potensi Sumber Daya Air di Indonesia Potensi Sumber Daya Air di Indonesia 1. *Seminar Nasional Aplikasi Teknologi Penyediaan Air Bersih Untuk Kabupaten/Kota Di Indonesia*, 1–20.

Statistik, B. P. (2016). Statistik Sumber Daya Laut dan Pesisir 2016. In *Badan Pusat Statistik*.

Statistik, B. P. (2017). Statistik Lingkngan Hidup Indonesia 2017. In *Badan Pusat Statistik*.

Suendri. (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem (Studi Kasus : UIN Sumatera Utara Medan). *Jurnal Ilmu Komputer Dan Informatika*, *3*(1), 1–9.

Supriyatna, A. (2018). Metode Extreme Programming Pada Pembangunan Web Aplikasi Seleksi Peserta Pelatihan Kerja. *Jurnal Teknik Informatika*, *11*(1), 1–18.

Yadav, K. S., Yasvi, M. A., & Shubhika. (2019). Review On Extreme Programming-XP. *International Conference on Robotics, Smart Technology and Electronics Engineering*.

JISKa, Vol. 4, No. 3, JANUARI, 2020, Pp. 185 – 192
ISSN : 2527 – 5836 (print) | 2528 – 0074 (online)

Journal Classification Based on Abstract Using Cosine Similarity and Support Vector Machine

Muhammad Habibi[(1)], Puji Winar Cahyo[(2)]
Program Studi Informatika
Universitas Jenderal Achmad Yani Yogyakarta
e-mail : muhammadhabibi17@gmail.com[(1)], pwcahyo@gmail.com[(2)]

Abstract

One of the problems related to journal publishing is the process of categorizing entry into journals according to the field of science. A large number of journal documents included in a journal editorial makes it difficult to categorize so that the process of plotting to reviewers requires a long process. The review process in a journal must be done planning according to the expertise of the reviewer, to produce a quality journal. This study aims to create a classification model that can classify journals automatically using the Cosine Similarity algorithm and Support Vector Machine in the classification process and using the TF-IDF weighting method. The object of this research is abstract in scientific journals. The journals will be classified according to the reviewer's field of expertise. Based on the experimental results, the Support Vector Machine method produces better performance accuracy than the Cosine Similarity method. The results of the calculation of the value of precision, recall, and f-score are known that the Support Vector Machine method produces better amounts, in line with the accuracy value.

Keywords : Text Mining, Cosine Similarity, Classification, Journal, Support Vector Machine

Abstrak

Salah satu masalah yang berkaitan dengan penerbitan jurnal yaitu proses pengkategorian jurnal masuk sesuai dengan bidang ilmu. Banyaknya jumlah dokumen jurnal yang masuk dalam suatu editorial jurnal membuatnya sulit untuk dilakukan pengkategorian sehingga proses plotting kepada *reviewer* membutuhkan proses yang lebih lama. Proses *review* pada suatu jurnal harus dilakukan plotting menyesuaikan dengan bidang keahlian dari *reviewer*, sehingga menghasilkan jurnal yang berkualitas. Penelitian ini bertujuan untuk membuat model klasifikasi yang dapat mengklasifikasikan jurnal secara otomatis menggunakan algoritma *Cosine Similarity* dan *Support Vector Machine* dalam proses pengklasifikasiannya dan menggunakan metode pembobotan TF-IDF. Objek penelitian ini adalah *abstract* pada jurnal ilmiah. Jurnal akan diklasifikasikan sesuai dengan rumpun ilmu bidang keahlian dari *reviewer*. Berdasarkan hasil eksperimen, metode *Support Vector Machine* menghasilkan akurasi performansi yang lebih baik dari pada metode *Cosine Similarity*. Hasil perhitungan nilai *precision*, *recall*, dan *f-score* diketahui bahwa metode *Support Vector Machine* menghasilkan nilai yang lebih baik, sejalan dengan nilai akurasi.

Kata Kunci : *Text Mining*, *Cosine Similarity*, Klasifikasi, Jurnal, *Support Vector Machine*

1. PENDAHULUAN

Perkembangan teknologi informasi membawa dampak yang sangat signifikan pada dunia Pendidikan. Salah satunya adalah melimpahnya informasi yang dapat diakses sebagai referensi dalam Pendidikan. Penyebaran jurnal atau artikel ilmiah sebagai bahan pendukung penelitian semakin meningkat. Dalam prosesnya, penerbitan jurnal memiliki beberapa tahapan mulai dari submission jurnal sampai jurnal tersebut terbit. Banyaknya jumlah dokumen jurnal yang masuk dalam suatu editorial jurnal membuatnya sulit untuk dilakukan pengkategorian sehingga proses *plotting* kepada *reviewer* membutuhkan proses yang lebih lama. Proses *review*

pada suatu jurnal harus dilakukan *plotting* menyesuaikan dengan bidang keahlian dari *reviewer*, sehingga menghasilkan jurnal yang berkualitas.

Untuk mempermudah mengkategorikan jurnal, diperlukan teknik pemrosesan teks yang dapat mengkategorikan sejumlah besar dokumen teks sesuai dengan tipenya, sehingga informasi yang tersedia dapat diakses dengan benar dan mudah diakses sesuai dengan kebutuhan pengguna. Salah satu pemecahan masalah dalam mengkategorikan dokumen teks dapat diselesaikan dengan menggunakan metode *text mining* yaitu klasifikasi.

Penelitian ini menggunakan algoritma *Cosine Similarity* untuk melakukan klasifikasi sesuai kemiripan teks *abstract* jurnal. *Cosine Similarity* telah banyak digunakan untuk melakukan pengklasifikasian teks seperti pengklasifikasian *tweet* populer (Ahmed, Razzaq, & Qamar, 2013), pengklasifikasian pertanyaan ujian (Jayakodi, Bandara, & Meedeniya, 2016), pengklasifikasian jawaban ujian (Saipech & Seresangtakul, 2018), pengklasifikasian komentar mahasiswa pada sistem evaluasi pembelajaran (Muhammad Habibi & Sumarsono, 2018) serta untuk pengklasifikasian dokumen *text* (Kadhim, Cheah, Ahamed, & Salman, 2014).

Selain algoritma *Cosine Similarity*, Penelitian ini juga menggunakan Algoritma *Support Vector Machine* (SVM) sebagai pembanding. SVM dikenal sebagai metode yang memiliki nilai akurasi yang sangat baik untuk pengklasifikasian data teks. Salah satu penerapan SVM untuk klasifikasi teks yaitu, klasifikasi judul skripsi (Hidayatullah & Maarif, 2016), dan klasifikasi komentar mahasiswa (Muhammad Habibi, 2017).

Tujuan penelitian ini adalah membuat sebuah model klasifikasi yang dapat mengklasifikasikan jurnal secara otomatis menggunakan algoritma *Cosine Similarity* dan *Support Vector Machine* dalam proses pengklasifikasiannya dan menggunakan metode pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF). Metode pembobotan menggunakan TF-IDF sudah banyak digunakan dalam pemrosesan data teks, seperti yang digunakan untuk pengolahan data *hashtag* pada *caption* Instagram (Muhamad Habibi & Cahyo, 2019) serta untuk analisis konten jejaring sosial twitter (Muhammad Habibi, 2018). Objek penelitian ini adalah *abstract* pada jurnal ilmiah. Jurnal akan diklasifikasikan sesuai dengan rumpun ilmu bidang keahlian dari *reviewer*. Sehingga diharapkan model klasifikasi yang dihasilkan pada penelitian ini dapat membantu meringankan kegiatan plotting *reviewer* pada editorial jurnal.

2. METODE PENELITIAN

2.1 DATASET

Data *abstract* jurnal yang digunakan dalam penelitian ini terdiri dari empat kelas bidang ilmu yaitu Sistem Cerdas, *Data Mining*, *Image Processing* dan Jaringan. Sebanyak 210 data jurnal yang digunakan dalam penelitian ini. Adapun detail *dataset* yang digunakan dapat dilihat pada Tabel 1.

Tabel 1. Jumlah Pengguna Internet di Indonesia

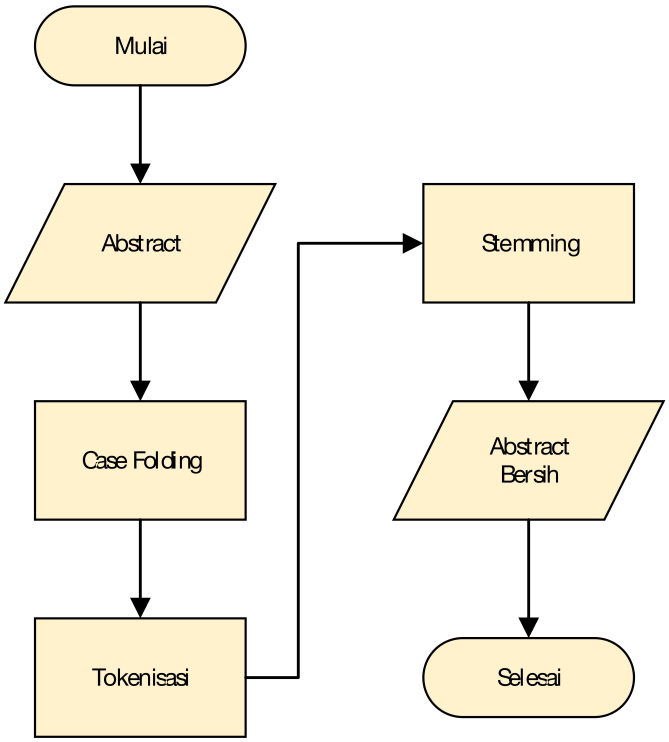
No	Kelas Bidang Ilmu	Jumlah Data
1	*Data Mining*	37
2	Sistem Cerdas	82
3	*Image Processing*	36
4	Jaringan	55
Total		210

2.2 PREPROCESSING

Penelitian ini terdiri dari beberapa tahapan, tahapan awal dalam penelitian ini adalah *preprocessing*. *preprocessing* data merupakan proses mempersiapkan dan membersihkan data teks sebelum teks dilakukan analisis (Haddi, Liu, & Shi, 2013). Adapun *flowchart preprocessing* dapat dilihat pada Gambar 1.

Tahapan *preprocessing* pada penelitian ini memiliki perbedaan dengan tahapan *preprocessing* yang dilakukan pada data teks media sosial. Data yang digunakan dalam penelitian ini merupakan data *abstract* bahasa Inggris yang terdapat pada jurnal. Berbeda dengan data teks sosial media, data teks *abtract* pada jurnal memiliki karakteristik kalimat yang baku dan sesuai dengan kaidah bahasa yang benar sehingga tidak terlalu banyak mengandung kata-kata tidak baku. Adapun tahapan *preprocessing* yang akan dilakukan dalam penelitian ini diantaranya adalah:

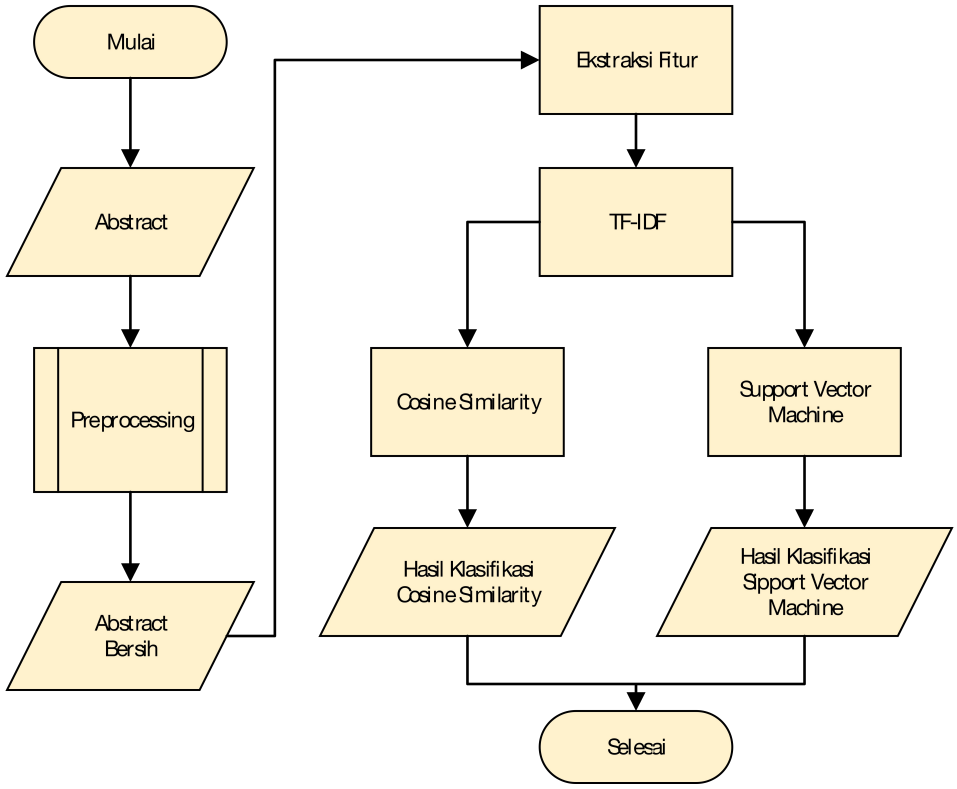
a. *Case Folding* yaitu proses untuk mengubah huruf kecil pada teks *abstract*.
b. Tokenisasi yaitu proses untuk membagi teks *abstract* ke dalam token.
c. *Stemming* yaitu proses mengubah kata berimbuhan menjadi kata dasar.



Gambar 1*. Flowchart Preprocessing*.

2.3 KLASIFIKASI

Tahapan selanjutnya setelah proses *preprocessing* adalah ekstraksi fitur, ekstraksi fitur bertujuan untuk mengidentifikasi entitas yang dirujuk (Siqueira & Barros, 2010). Pada penelitian ini, fitur yang digunakan adalah *Term Frequency – Inverse Document Frequency* (TF-IDF). TF-IDF terdiri dari dua buah nilai komponen yaitu *term-frequency* dan *inverse document frequency*. Skema pembobotan TF-IDF memberikan bobot *term* dalam suatu dokumen (Manning, Raghavan, & Schutze, 2009). Setelah didapatkan nilai TF-IDF, langkah selanjutnya adalah proses klasifikasi menggunakan *Cosine Similarity* dan *Support Vector Machine*. Flowchart proses klasifikasi dapat dilihat pada Gambar 2.



Gambar 2. Flowchart proses klasifikasi.

2.4 METODE EVALUASI

Estimasi tingkat kesalahan prediksi diperlukan untuk mengevaluasi kinerja model klasifikasi yang sudah dibuat. *Cross validation* dapat digunakan untuk memperkirakan kesalahan prediksi (Fushiki, 2011). Dalam pendekatan *cross validation*, setiap *record* digunakan beberapa kali dalam jumlah yang sama untuk pelatihan dan untuk pengujian.

Metode *k-fold cross-validation* mensegmentasi data ke dalam *k* partisi berukuran sama. Pada metode ini salah satu dari partisi dipilih untuk pengujian, sedangkan sisanya digunakan untuk pelatihan. Prosedur ini diulang sebanyak *k* kali sehingga setiap partisi digunakan untuk pengujian tepat satu kali. Total *error* ditentukan dengan menjumlahkan *error* untuk semua *k* proses tersebut. (Muhammad Habibi, 2017).

Perhitungan validasi hasil klasifikasi dapat diukur menggunakan *precision, recall*, dan *harmonic mean* dari *precision* dan *recall* yakni *F-score* (Dermawan, 2016). Pengujian dengan *precission* dan *recall* pada suatu entitas menunjukan hasil yang baik sehingga dapat meningkatkan nilai *F-score* (Cahyo, 2017). *Precision* merupakan persentase model klasifikasi dapat melakukan pelabelan benar dari label yang dikenali. *Recall* merupakan persentase seberapa banyak label dapat dikenali oleh model klasifikasi. Sedangkan *F-score* merupakan penghitungan evaluasi temu kembali informasi yang mengkombinasikan *recall* dan *precision*.

3. HASIL DAN PEMBAHASAN

3.1 HASIL AKURASI MODEL KLASIFIKASI

Proses perhitungan akurasi dilakukan dalam 10 kali percobaan menggunakan *K-fold cross validation*. Hasil perhitungan akurasi dapat dilihat pada Tabel 2.

Tabel 2. Hasil Perhitungan Akurasi Model Klasifikasi

Percobaan	Akurasi	
	Cosine Similarity	***Support Vector Machine***
1	0,81	0,71
2	0,62	0,76
3	0,67	0,57
4	0,67	0,81
5	0,57	0,67
6	0,67	0,76
7	0,57	0,67
8	0,62	0,76
9	0,48	0,81
10	0,48	0,95
Rata-rata	**0,61**	**0,75**

Berdasarkan Tabel 2, Hasil percobaan menunjukkan bahwa model klasifikasi yang dibangun dengan menggunakan algoritma *Cosine Similarity* memiliki nilai akurasi rata-rata dalam 10 kali percobaan yaitu 61%. Sementara itu, medel klasifikasi yang dibangun menggunakan algoritma *Support Vector Machine* memiliki nilai akurasi rata-rata dalam 10 kali percobaan yaitu 75%. Pada penelitian ini didapatkan bahwa akurasi algoritma *Support Vector Machine* memiliki akurasi yang lebih baik dibandingkan dengan algoritma *Cosine Similarity*.

3.2 PERHITUNGAN *PRECISION*, *RECALL* DAN *F-SCORE*

Hasil perhitungan evaluasi *precision, recall* dan *f-score* menggunakan algoritma *Cosine Similarity* dapat dilihat pada Tabel 3.

Tabel 3. Hasil Perhitungan *Precision, Recall* dan *F-score Cosine Similarity*

Percobaan	***Precision***	***Recall***	***F-score***
1	0,86	0,88	0,85
2	0,73	0,67	0,61
3	0,68	0,74	0,62
4	0,81	0,62	0,67
5	0,68	0,54	0,50
6	0,64	0,73	0,66
7	0,63	0,63	0,56
8	0,53	0,52	0,51
9	0,44	0,43	0,43
10	0,47	0,46	0,43
Rata-rata	**0,65**	**0,62**	**0,58**

Berdasarkan hasil yang ditunjukkan pada Tabel 3, diketahui bahwa algoritma *Cosine Similarity* menghasilkan nilai rata-rata *precision* sebesar 65%, *recall* sebesar 63%, sedangkan *f-score*

yang dihasilkan adalah 58%. Hasil perhitungan evaluasi menggunakan algoritma *Support Vector Machine* dapat dilihat pada Tabel 4. Pada tabel tersebut, dapat diketahui bahwa hasil nilai rata-rata *precision* sebesar 72%, *recall* sebesar 69%, sedangkan *f-score* sebesar 67%.

Tabel 4. Hasil Perhitungan *Precision, Recall* dan *F-score* SVM

Percobaan	*Precision*	*Recall*	*F-score*
1	0,72	0,71	0,71
2	0,56	0,58	0,56
3	0,59	0,51	0,47
4	0,89	0,78	0,81
5	0,37	0,45	0,40
6	0,88	0,75	0,76
7	0,67	0,67	0,64
8	0,78	0,69	0,70
9	0,86	0,78	0,75
10	0,92	0,97	0,94
Rata-rata	**0,72**	**0,69**	**0,67**

Hasil perhitungan *precision*, *recall* dan *f-score* dari algoritma *Cosine Similarity* dan *Support Vector Machine* menunjukkan bahwa hasil pengujian model klasifikasi yang dibangun menggunakan *Support Vector Machine* memiliki hasil *precision*, *recall* dan *f-score* lebih baik dibandingkan dengan *Cosine Similarity*.

3.3 PERHITUNGAN *PRECISION, RECALL* DAN *F-SCORE* UNTUK TIAP LABEL

Hasil perhitungan *precision*, *recall* dan *f-score* untuk masing-masing *class label* menggunakan *Cosine Similarity* dan *Support Vector Machine* secara berturut-turut dapat dilihat pada Tabel 5 dan Tabel 6.

Tabel 5. Hasil *Precision, Recall* dan *F-score* tiap label menggunakan *Cosine Similarity*

Class Label	*Precision*	*Recall*	*F-score*
Sistem Cerdas	0,60	0,76	0,65
Data Mining	0,71	0,53	0,54
Image Processing	0,50	0,51	0,45
Jaringan	0,78	0,68	0,70

Tabel 6. Hasil *Precision*, *Recall* dan *F-score* tiap label menggunakan SVM

Class Label	*Precision*	*Recall*	*F-score*
Sistem Cerdas	0,67	0,89	0,75
Data Mining	0,64	0,46	0,50
Image Processing	0,69	0,63	0,65
Jaringan	0,89	0,77	0,81

Berdasarkan hasil Tabel 5, hasil *Precision, Recall* dan *F-score* untuk setiap label menggunakan *Cosine Similarity* didapatkan bahwa kategori *Image Processing* memiliki nilai *precision*, *recall* dan *f-score* paling rendah dibandingkan dengan kategori yang lain. Sedangkan hasil *Precision*, *Recall* dan *F-score* untuk setiap label menggunakan *Support Vector Machine* didapatkan bahwa kategori *Data Mining* memiliki nilai *precision, recall* dan *f-score* paling rendah dibandingkan dengan kategori yang lain.

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, maka diperoleh kesimpulan bahwa, penelitian ini berhasil membuat model yang dapat mengklasifikasikan jurnal secara otomatis menggunakan algoritma *Cosine Similarity* dan *Support Vector Machine* dan menggunakan metode pembobotan TF-IDF. Hasil akurasi pengujian metode *Cosine Similarity* diperoleh sebesar 61% sedangkan metode *Support Vector Machine* didapatkan akurasi sebesar 75%. Metode *Support Vector Machine* menghasilkan akurasi performansi yang lebih baik dari pada metode *Cosine Similarity*. Hasil perhitungan nilai *precision*, *recall*, dan *f-score* diketahui bahwa metode *Support Vector Machine* menghasilkan nilai yang lebih baik, sejalan dengan nilai akurasi.

.

DAFTAR PUSTAKA

Ahmed, H., Razzaq, M. A., & Qamar, A. M. (2013). Prediction of popular tweets using Similarity Learning. *ICET 2013 - 2013 IEEE 9th International Conference on Emerging Technologies*. https://doi.org/10.1109/ICET.2013.6743524

Cahyo, P. W. (2017). *Model Monitoring Sebaran Penyakit Demam Berdarah di Indonesia Berdasarkan Analisis Pesan Twitter*. Universitas Gadjah Mada Yogyakarta.

Dermawan, R. (2016). *Klasifikasi Tweet dan Pengenalan Entitas Bernama pada Tweet Bencana Dengan Support Vector Machine*. Universitas Gadjah Mada.

Fushiki, T. (2011). Estimation of prediction error by using K-fold cross-validation. *Statistics and Computing*, *21*(2), 137–146. https://doi.org/10.1007/s11222-009-9153-8

Habibi, Muhamad, & Cahyo, P. W. (2019). Clustering User Characteristics Based on the influence of Hashtags on the Instagram Platform. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, *13*(4), 399–408.

Habibi, Muhammad. (2017). *Analisis Sentimen dan Klasifikasi Komentar Mahasiswa pada Sistem Evaluasi Pembelajaran Menggunakan Kombinasi KNN Berbasis Cosine Similarity dan Supervised Model*. *Departemen Ilmu Komputer dan Elektronika, Fakultas Matematika dan Ilmu Pengetahuan Alam*. Universitas Gadjah Mada.

Habibi, Muhammad. (2018). Analisis Konten Jejaring Sosial Twitter dalam Kasus Pemilihan Gubernur DKI 2017. *Teknomatika*, *11*(1), 31–40.

Habibi, Muhammad, & Sumarsono. (2018). Implementation of Cosine Similarity in an automatic classifier for comments. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *3*(2), 38–46.

Haddi, E., Liu, X., & Shi, Y. (2013). The Role of Text Pre-processing in Sentiment Analysis. *Procedia Computer Science*, *17*, 26–32. https://doi.org/10.1016/j.procs.2013.05.005

Hidayatullah, A. F., & Maarif, M. R. (2016). Penerapan Text Mining dalam Klasifikasi Judul Skripsi. In *Seminar Nasional Aplikasi Teknologi Informasi (SNATi) Agustus* (pp. 1907–5022). Yogyakarta.

Jayakodi, K., Bandara, M., & Meedeniya, D. (2016). An automatic classifier for exam questions with WordNet and Cosine similarity. *2nd International Moratuwa Engineering Research Conference, MERCon 2016*, 12–17. https://doi.org/10.1109/MERCon.2016.7480108

Kadhim, A. I., Cheah, Y. N., Ahamed, N. H., & Salman, L. A. (2014). Feature extraction for co-occurrence-based cosine similarity score of text documents. *2014 IEEE Student Conference on Research and Development, SCOReD 2014*, 2–5.

https://doi.org/10.1109/SCORED.2014.7072954

Manning, C. D., Raghavan, P., & Schutze, H. (2009). *An Introduction to Information Retrieval*. Cambridge, England: Cambridge University Press. https://doi.org/10.1109/LPT.2009.2020494

Saipech, P., & Seresangtakul, P. (2018). Automatic Thai Subjective Examination using Cosine Similarity. *ICAICTA 2018 - 5th International Conference on Advanced Informatics: Concepts Theory and Applications*, 214–218. https://doi.org/10.1109/ICAICTA.2018.8541276

Siqueira, H., & Barros, F. (2010). A Feature Extraction Process for Sentiment Analysis of Opinions on Services. *Proceedings of the III International Workshop on Web and Text Intelligence (WTI)*.

JIKa, Vol. 4, No. 3, JANUARI, 2020, Pp. 193 – 201
ISSN : 2527 – 5836 (print) | 2528 – 0074 (online)

Diagnosa Penyakit Demam Berdarah Dengue (DBD) menggunakan Metode Learning Vector Quantization (LVQ)

Firman Tawakal [(1)], Ahmedika Azkiya [(2)]
Jurusan Teknik Informatika[(1)], Manajemen Informatika[(2)]
[(1)]Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Dumai,
[(2)]Akademi Manajemen Informatika dan Komputer (AMIK) Dumai
Jl. Utama Karya Bukit Batrem Dumai - Riau
e-mail : firman.tawakal@gmail.com[(1)], ahmedikaazkiya@gmail.com[(2)]

Abstract

Dengue Hemorrhagic Fever is a disease that is carried and transmitted through the mosquito Aedes aegypti and Aedes albopictus which is commonly found in tropical and subtropical regions such as in Indonesia to Northern Australia. in 2013 there are 2.35 million reported cases, which is 37,687 case is heavy cases of DHF. DHF's symthoms have a similarity with typhoid fever, it often occur wrong handling. Therefore we need a system that is able to diagnose the disease suffered by patients, so that they can recognize whether the patient has DHF or Typhoid. The system will be built using Neural Network Learning Vector Quantization (LVQ) based on the best training results. This research is to diagnose Dengue *Hemorrhagic Fever using LVQ with input parameters are hemoglobin, leukocytes, platelets, and heritrocytes. Based on result, the best accuracy is 97,14% with Mean Square Error (MSE) is 0.028571 with 84 train data and 36 test data. Conclution from the research is LVQ method can diagnose DHF*

Keywords*: Dengue Hemorrhagic Fever; Learning Vector Quantization; classification; Neural Network;*

Abstrak

Demam Berdarah Dengue (DBD) merupakan penyakit yang dibawa dan ditularkan melalui nyamuk Aedes aegypti dan Aedes albopictus. Gejala yang dialami oleh penderita DBD ternyata memiliki kemiripan dengan gejala penyakit Typhoid, sehingga seringkali terjadi salah penanganan. Untuk itu diperlukan suatu sistem yang mampu mendiagnosa penyakit yang diderita oleh pasien, sehingga mampu mengenali apakah pasien menderita penyakit DBD atau Typhoid. Sistem yang akan dibangun menggunakan metode Jaringan Syaraf Tiruan *Learning Vector Quantization* (LVQ). Penelitian yang dilakukan yaitu mendiagnosa penyakit Demam Berdarah Dengue menggunakan metode LVQ dengan parameter input yaitu hemoglobin, leukosit, trombosit, dan heritrosit. Berdasarkan hasil yang diperoleh, nilai akurasi terbaik adalah 97,14% dengan nilai *Mean Square Error* (MSE) sebesar 0.028571 dengan jumlah data latih 84 dan data uji berjumlah 36. Kesimpulan dari penelitian yang dilakukan adalah Metode LVQ mampu melakukan diagnosa penyakit DBD dengan baik.

Kata Kunci: Demam Berdarah Dengue; *Learning Vector Quantization*; klasifikasi; Jaringan Syaraf Tiruan;

1. PENDAHULUAN

Demam berdarah Dengue (DBD) merupakan wabah yang menyerang berbagai negara secara global, dengan lebih dari 500.000 kasus dilaporkan setiap tahun. Penyakit DBD sebagian besar ditularkan oleh nyamuk Aedes aegypti dan Aedes albopictus [1]. Gejala yang dialami oleh penderita DBD ternyata memiliki kemiripan dengan gejala penyakit lain yaitu Tifus / typhoid. Bagi orang yang awam akan mengakibatkan kebingungan dan salah diagnosa jika tidak benar – benar diperiksa secara teliti. Sehingga seringkali terjadi salah penanganan yang mengakibatkan efek dari obat yang diberikan tidak akan berpengaruh pada penyakit. Untuk itu diperlukan suatu sistem yang mampu mendiagnosa penyakit yang diderita oleh pasien, sehingga mampu mengenali apakah pasien menderita penyakit DBD atau Typhoid.

Beberapa penelitian terkait yang menggunakan metode LVQ yaitu Rosario, dkk (2018) yang melakukan penelitian mengenai Penerapan Jaringan Syaraf Tiruan untuk klasifikasi penyakit demam berdarah dengue (Rosario, 2018). Penelitian yang berjudul *A Study on the Application of Learning Vector Quantization Neural Network in Pattern Classification*, yang menyatakan

bahwa metode LVQ memiliki kecepatan pelatihan jaringan dan memiliki kebutuhan yang lebih sedikit untuk sample data dan jumlah layer kompetisi. Selain itu metode LVQ lebih efektif dalam pengenalan pola klasifikasi (Shuo, 2014). Selanjutnya pada penelitian yang berjudul desain Optimal Modular LVQ untuk klasifikasi Arrythmia berdasarkan variabel data training dan data test yang melakukan penelitian untuk membandingkan pengaruh jumlah data training dan testing terhadap akurasi klasifikasi pada LVQ (Amezcua, 2015). Selanjutnya yaitu penelitian tentang pengenalan citra pose tangan menggunakan LVQ yang menghasilkan akurasi klasifikasi sebesar 99% (Felice, 2018). Ghanem dkk (2016) melakukan penelitian mengenai klasifikasi hadits menggunakan LVQ untuk klasifikasi hadits sahih, Da'if dan maudu'. Budianita dan Novriyanto (2015) yang melakukan penelitian tentang Klasifikasi Status Gizi Balita Berdasarkan Indikator Antropometri Berat Badan Menurut Umur Menggunakan *LVQ*. Sintawati (2016) yang membahas penelitian mengenai Diagnosa penyakit Demam Berdarah Dengue (DBD) dengan Algoritma pembelajaran *hybrid* dan *backpropagation* Berbasis *neural network.* Leleuri, dkk (2016) yang membandingkan metode *Backpropagation* dengan metode LVQ dengan judul penelitian yaitu Sistem Diagnosa Penyakit Dalam dengan Menggunakan Jaringan Saraf Tiruan Metode *Backpropagation* dan LVQ*.*

Pada penelitian ini dilakukan penelitian tentang diagnosa DBD menggunakan jaringan syaraf tiruan LVQ. Diagnosa yang diberikan akan diklasifikasi ke dalam DBD atau Tifus. Hasil klasifikasi dibandingkan dengan hasil yang sebenarnya untuk mendapatkan nilai akurasi penelitian. Data yang digunakan adalah data pasien DBD dan Typhoid pada tahun 2017 sampai 2018 pada Rumah Sakit Umum Daerah Kota Dumai.

2. TINJAUAN PUSTAKA

2.1 Metode *Learning Vector Quantization* (LVQ)

Learning Vector Quantization (LVQ) adalah suatu metode yang digunakan untuk melakukan pelatihan terhadap lapisan – lapisan kompetitif yang terawasi. Lapisan kompetitif akan belajar secara otomatis untuk melakukan klasifikasi terhadap vektor input yang diberikan. Vektor – vektor input akan dikelompokkan dalam kelas yang sama apabila beberapa vektor input memiliki jarak yang sangat berdekatan (Kusumadewi, 2004). Diasumsikan bahwa serangkaian pola pelatihan dengan klasifikasi yang tersedia bersama dengan distribusi awal vektor referensi. Setelah pelatihan, kelas yang sama akan ditugaskan untuk melakukan klasifikasi vektor masukan sebagai unit keluaran, sedangkan yang mempunyai vektor referensi diklasifikasikan sebagai vektor masukan.

Learning Vector Quantization (LVQ) merupakan jaringan lapisan tunggal (*single-layer net*) di mana lapisan masukan terkoneksi secara langsung dengan setiap *neuron* pada keluaran. Koneksi antar *neuron* tersebut dihubungkan dengan bobot/*weight.* Bobot merupakan nilai matematis dari koneksi yang mentransfer data dari satu lapisan ke lapisan lainnya, yang berfungsi untuk mengatur jaringan sehingga dapat menghasilkan *output* yang diinginkan. Bobot pada LVQ sangat penting, karena dengan bobot ini *input* dapat melakukan pembelajaran dalam mengenali suatu pola. Vektor bobot berfungsi untuk menghubungkan setiap *neuron* pada lapisan *input* dengan masing-masing *neuron* pada lapisan *output*. Vektor bobot biasanya dituliskan dengan *wtj=(wt1,wt2,wt3,…wtm)* di mana *t* menunjukkan kelas yang nilainya antara 1 sampai K, dengan K adalah banyaknya kelas pada lapisan *output*, sedangkan *m* adalah banyaknya variabel yang digunakan. Kelebihan dari LVQ adalah:

1. Nilai *error* yang lebih kecil dibandingkan dengan Jaringan Saraf Tiruan seperti *Backpropagation*.
2. Dapat meringkas data set yang besar menjadi vektor *codebook* berukuran kecil untuk klasifikasi.
3. Dimensi dalam *codebook* tidak dibatasi seperti dalam teknik *nearest neighbour*.
4. Model yang dihasilkan dapat diperbaharui secara bertahap.

2.2 Demam Berdarah Dengue (DBD)

Demam Berdarah Dengue merupakan penyakit yang berasal dari virus Dengue yang dibawa dan ditularkan melalui gigitan nyamuk *Aedes albopictus / Aedes aegypti* yang memenuhi kriteria WHO untuk DBD. Penyakit DBD menjadi salah satu penyakit yang menjadi perhatian serius untuk ditangani oleh pemerintah Indonesia. Hal ini di sebabkan karena penyebaran penyakit yang tiap tahunnya semakin meluas. Penderita penyakit DBD memiliki ciri demam tinggi mendadak disertai manifestasi pendarahan dan bertendensi mengalami renjatan (*shock*) dan kematian (Depkes RI, 2010).

Tanda maupun gejala penderita DBD sifatnya tidak khas, artinya bahwa tanda dan gejala yang ditimbulkan dapat bervariasi tergantung pada penderita berdasarkan derajat yang dialaminya. Pada umumnya tanda – tanda atau gejala yang ditimbulkan oleh penderita DBD adalah sebagai berikut (Dini, dikutip oleh Sintawati, 2016) :

a. Mengalami demam tinggi
b. Mengalami perdarahan atau bintik merah pada kulit
c. Mengalami keluhan pada saluran pernafasan
d. Mengalami keluhan pada saluran pencernaan
e. Biasanya merasakan sakit saat menelan
f. Mengalami keluhan pada bagian tubuh yang lain, seperti nyeri otot, tulang, sendi, dan ulu hati, serta pegal – pegal di seluruh tubuh.
g. Mengalami pembesaran hati, limpa, dan kelenjar getah bening, yang akan kembali normal pada masa penyembuhan.

Pada kondisi parah, penderita akan mengalami keadaan renjatan (*shock*), yang dikenal dengan *Dengue Shock Syndrome* (DSS), dengan tanda – tanda sebagai berikut:

a. Kulit terasa lembab dan dingin.
b. Tekanan darah menurun.
c. Denyut nadi cepat dan lemah.
d. Mengalami nyeri perut yang hebat.
e. Mengalami pendarahan, baik dari mulut, hidung, maupun anus.
f. Lemah dan mengalami penurunan tingkat kesadaran.
g. Mengalami kegelisahan.
h. Mulut, hidung, dan ujung jari penderita tampak kebiru – biruan.
i. Tidak buang air kecil selama 4-6 jam.

Hasil laboratorium umumnya dilakukan untuk mempertegas diagnosa dokter untuk penyakit DBD atau *Typhoid*. Untuk membedakan pasien terdiagnosa DBD atau *typhoid* dapat dilihat dari pola demam yang di derita pasien serta hasil laboratorium. Selain itu untuk kriteria cek darah di laboratorium dengan status darah normal dapat dilihat dari data berikut :

a. *Trombosit* : 150.000 – 400.000 /cmm
b. *Hemoglobin* : L: 14,0-18,0 P: 12,0-18,0 gr/dl
c. *Hematokrit* : L: 42-52 P: 37-47 %
d. *Leukosit* : 4.800-10.800 uL

2.3 Demam *Typhoid*

Demam *Typhoid* atau yang lebih dikenal dengan demam Tifus adalah penyakit yang seringkali dikelirukan dengan penyakit DBD dikarenakan gejala demam yang hampir sama. Adapun perbedaan yang dapat dilihat adalah pola kenaikan demam yang terjadi. Pada DBD seringkali demam mendadak tinggi dalam 2 hari awal dan menurun pada hari ke 3 sampai hari ke 5. Namun berbeda dengan demam *Typhoid*, pasien akan mengalami demam yang meningkat sangat tinggi setelah 3 sampai 5 hari (Dini, dikutip oleh Sintawati, 2016).

3. METODE PENELITIAN

Secara singkat tujuan dari penelitian ini adalah untuk melakukan klasifikasi atau pengenalan pola dari penyakit DBD dengan menggunakan data laboratorium. Selanjutnya data yang diperoleh akan dilatih menggunakan metode Jaringan Syaraf Tiruan LVQ. Sehingga setelah dilatih diharapkan metode LVQ mampu mengenali pola dan melakukan klasifikasi penyakit DBD secara otomatis. Dalam proses pelatihan data menggunakan metode LVQ, terdapat tahapan – tahapan yang akan dilakukan yaitu:

A. Menentukan tujuan sistem yaitu mampu mengenali pola pelatihan sehingga mampu melakukan klasifikasi apakah pasien terdiagnosa DBD atau *Typhoid* berdasarkan data laboratorium pasien.
B. Mengambil data darah hasil laboratorium Rumah Sakit Umum Daerah Kota Dumai pada tahun 2017 sampai 2018.
C. Merancang jaringan LVQ melalui beberapa tahapan sebagai berikut:
   i. Menentukan data latih dan data uji dengan perbandingan 86 data latih dan 34 data uji. Dengan total data 120.
   ii. Menganalisa data yang telah diperoleh
   iii. Menentukan parameter yang akan digunakan untuk proses pelatihan. Algoritma LVQ memiliki beberapa parameter pelatihan seperti *input, output*, *learning rate* (α), *goal, epoch*, dan *neuron hidden layer.* Selanjutnya akan dicoba kombinasi parameter yang menghaslikan akurasi klasifikasi tertinggi. Algoritma pembelajaran LVQ adalah [10]:
      – Tetapkan bobot awal variabel input ke-j menuju ke kelas ke-I (Wij), maksimum epoch, parameter learning rate (α), pengurangan learning rate (Decα), dan minimal learning rate (Minα).
      – Masukkan data input (Xij) dan target berupa kelas (Tk)
      – Tetapkan kondisi awal (epoch=0)
      – Kerjakan jika (epoch ≤ MaxEpoch) dan (α ≥ Minα):
      – Epoch = epoch+1
      – Kerjakan untuk i=1 sampai n
      – Tentukan J sedemikian hingga |Xi-Wj| minimum.
      – Perbaiki Wj dengan ketentuan:
         a. Jika T = Cj maka Wj(baru) = Wj (lama) + α (Xi-Wj(lama)) (1)
         b. Jika T ≠ Cj maka Wj(baru) = Wj (lama) - α (Xi-Wj(lama)) (2)
   iv. Kurangi nilai α (dilakukan dengan: α = α – Decα atau α = α * Decα)
D. Implemetasi dan pengujian jaringan LVQ dengan melakukan uji coba klasifikasi menggunakan data yang telah disediakan. Pengujian akan menghasilkan presentase akurasi dan nilai *error* yang diukur dengan menggunakan nilai *Mean Square Error* (MSE). Adapun rumus untuk menghitung akurasi adalah sebagai berikut:

$$Akurasi = \frac{\sum pengujian\ bernilai\ benar}{\sum Banyak\ data\ uji} x\ 100\% \quad (3)$$

Sedangkan untuk menghitung nilai MSE dirumuskan sebagai berikut:

$$MSE = \sum \frac{(Pt-At)^2}{n} \quad (4)$$

3.1. Rancangan Jaringan Syaraf Tiruan LVQ

Variabel masukan yang digunakan oleh jaringan syaraf tiruan LVQ untuk melakukan diagnosa penyakit DBD berupa data darah dari laboratorium Rumah Sakit Umum Daerah Kota Dumai. Adapun variabel masukan yang digunakan dapat dilihat dalam Tabel 1. Data yang tidak berupa data numerik akan di ubah ke bentuk numerik.

Tabel 1. Keterangan Variabel Masukan

Variabel	Keterangan	Nilai Input
X1	*Hemoglobin*	Bilangan Desimal
X2	*Leukosit*	Bilangan Bulat
X3	*Trombosit*	Bilangan Bulat
X4	*Hematokrit*	Bilangan Bulat
X5	Jenis Kelamin	0 = Perempuan 1 = Laki - Laki

Pada metode LVQ target / kelas output harus ditentukan terlebih dahulu. Output yang dihasilkan adalah klasifikasi diagnosa apakah pasien terdiagnosa DBD atau Typhoid. Data input terlebih dahulu harus di normalisasi ke dalam nilai dengan kisaran 0 dan 1 agar sistem dapat melakukan klasifikasi dengan lebih cepat. Rumus untuk normalisasi adalah:

$$X' = \frac{X - min(X)}{max(X) - min(X)} \quad (5)$$

dimana :

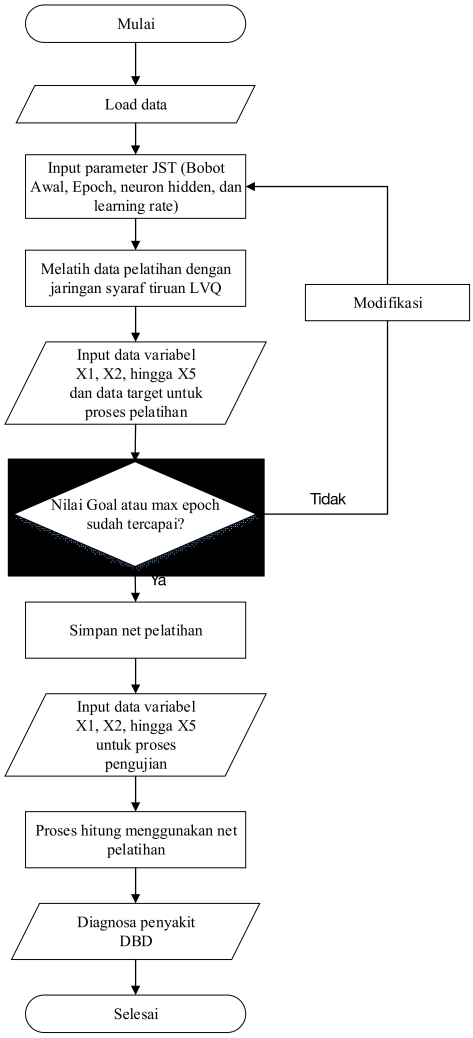
x' : Hasil transformasi data

xmax : Nilai terbesar

xmin : Nilai terkecil

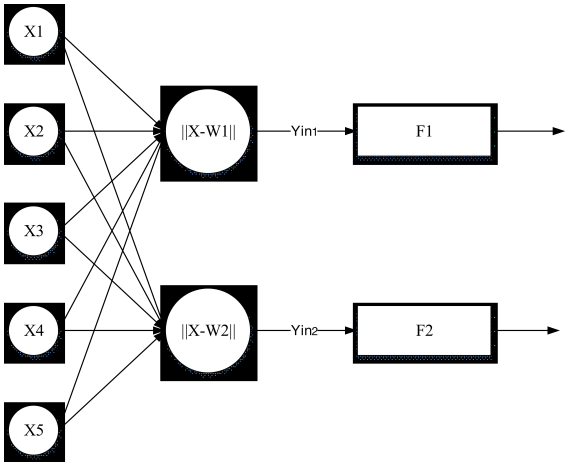
3.2. Perancangan Proses Pelatihan

Proses pelatihan sebagai bagian awal dari sistem diagnosa penyakit DBD. Ada tahapan yang perlu dilakukan yaitu melakukan *input* data – data yang sudah diolah yakni data latih dan data uji yang akan digunakan oleh sistem. Data masukan diubah terlebih dahulu dari format *string* ke *integer* berdasarkan ketentuan yang telah ditetapkan sebelumnya. Sedangkan untuk data target, penyakit akan dikelompokkan ke dalam kelompok penyakit DBD dan Typhoid. Untuk lebih jelasnya mengenai tahap – tahap dalam diagnosa penyakit DBD dapat dilihat pada *flowchart* berikut.



Gambar 1. *Flowchart* Sistem Diagnosa Penyakit DBD Menggunakan Metode LVQ

berdasarkan perancangan jaringan yang telah dibuat, maka arsitektur jaringan LVQ dapat digambarkan seperti pada gambar 2 berikut:



Gambar 2. Arsitektur Jaringan Syaraf Tiruan LVQ Diagnosa Penyakit DBD

4. HASIL DAN PEMBAHASAN

Pada sub bab ini akan dibahas hasil dari sistem yang telah dirancang dan dibuat. Pembahasan yang dilakukan yakni mengenai hasil arsitektur jaringan syaraf tiruan serta akurasi yang didapat dalam proses pelatihan data dan dalam memklasifikasi hasil diagnosa DBD. Uji coba yang akan dilakukan yakni dengan mengubah dan mencoba berbagai kombinasi parameter jaringan syaraf tiruan LVQ seperti learning rate, jumlah neuron hidden layer, dan fungsi pelatihan lain yang kemudian akan diavariasikan untuk melihat pengaruh dari perubahan parameter dan memilih mana susunan arsitektur jaringan syaraf tiruan yang menghasilkan akurasi terbaik.

4.1. Pengumpulan Data

Pada penelitian ini, dibuat sistem diagnosa penyakit Demam Berdarah Dengue pada Rumah Sakit Umum Daerah Kota Dumai. Untuk membuatnya diperlukan data – data terkait data pasien yang telah didiagnosa oleh dokter dan mendapat hasil laboratorium positif DBD dan pasien dengan hasil laboratorium positif Typhoid. Data yang didapat yaitu data pasien pada tahun 2017 hingga tahun 2018. Setelah diperoleh data pasien, maka selanjutnya data - data tersebut digunakan sebagai data latih dan data uji serta target yang digunakan oleh sistem sebagai pembelajaran.

4.2. Proses Pelatihan

Adapun tahapan – tahapan pada proses pelatihan menggunakan metode LVQ yaitu:

a. Memasukkan data yang telah di normalisasi yang akan digunakan pada proses pelatihan sebesar 84 data sesuai dengan yang telah penulis tentukan diawal.

Tabel 2. Sampel Data Latih

No	Hemoglobin	Leukosit	Trombosit	Hematokrit	Jenis Kelamin	Diagnosa
1	18.3	3000	101000	50	Laki - laki	DBD
2	13.6	1100	51000	37	Laki - laki	DBD
3	12.2	3900	94000	33	Perempuan	DBD
...	...	...	...	...	...	...
83	12.2	5100	172000	33	Laki - laki	TYP
84	12.8	3000	168000	36	Perempuan	TYP

b. Langkah selanjutnya adalah melakukan pelatihan pada data yang telah dipilih. Sebelum memulai pelatihan, terlebih dahulu harus mengisi data yang dibutuhkan untuk kriteria pelatihan, yaitu *jumlah epoch, learning rate, goal,* dan jumlah *neuron hidden layer.* Pada proses pelatihan ini akan dilakukan berbagai kombinasi kriteria pelatihan. Proses akan dilakukan berulang kali untuk mendapatkan hasil klasifikasi dengan akurasi tertinggi.

c. Setelah menemukan hasil pelatihan dengan akurasi tertinggi, maka hasil pelatihan akan disimpan untuk dilakukan klasifikasi, maka langkah berikutnya yaitu melakukan diagnosa penyakit DBD yang menggunakan beberapa parameter yang telah ditentukan yaitu *hemoglobin, leukosit, trombosit, hematokrit*, dan jenis kelamin dari data uji yang telah disiapkan.

Tabel 3. Sampel Data Uji

No	Hemoglobin	Leukosit	Trombosit	Hematokrit	Jenis Kelamin	Diagnosa
1	12.9	3100	169000	37	Laki - laki	TYP
2	9.7	8000	141000	27	Perempuan	TYP
3	13.7	17000	246000	39	Laki - laki	TYP
...	...	...	...	...	...	...
35	10.6	4700	81000	33	Perempuan	DBD
36	13.8	3100	77000	43	Perempuan	DBD

4.3. Analisa Hasil Klasifikasi

Hasil dari pelatihan yang telah dilakukan menggunakan data pasien RSUD Kota Dumai tahun 2017 dan 2018 yang dilakukan uji coba langsung untuk mengklasifikasi hasil diagnosa penyakit DBD. Data yang digunakan untuk memklasifikasi hasil diagnosa DBD adalah sebanyak 120 data yang terbagi ke dalam 84 data latih dan 36 data uji.

Proses pengujian dilakukan untuk melihat parameter mana yang paling mempengaruhi tingkat akurasi proses pelatihan. adapun kriteria yang dirubah untuk melihat hasil klasifikasi terbaik yaitu learning *rate* dan jumlah *neuron hidden layer*. Sedangkan untuk *goal* ditetapkan 0.1. berikut adalah hasil uji coba dari berbagai kombinasi Jaringan Syaraf Tiruan LVQ yang disajikan dalam Tabel 4.

Tabel 4. Hasil Klasifikasi diagnosa Demam Berdarah Dengue

No	*Final epoch*	*Learning Rate*	*Neuron hidden layer*	Klasifikasi Salah	Klasifikasi Berhasil	MSE	Akurasi
1	17	0.1	10	1	34	0.028571	97.22 %
2	28	0.2	10	1	34	0.028571	97.22 %
3	20	0.3	10	1	34	0.028571	97.22 %
4	25	0.4	10	2	33	0.055556	94.44 %
5	9	0.5	10	2	33	0.057143	94.44 %
6	20	0.1	20	1	34	0.028571	97.22 %
7	30	0.2	20	2	33	0.057143	94.44 %
8	11	0.3	20	1	34	0.028571	97.22 %
9	12	0.4	20	2	33	0.057143	94.44 %
10	10	0.5	20	2	33	0.057143	94.44 %

Dari Tabel 4 dapat dilihat bahwa nilai akurasi tertinggi adalah sebesar 97.22% dengan MSE 0.028571. Dari hasil pengujian terlihat bahwa akurasi yang dihasilkan dari kombinasi jaringan tidak mutlak ditentukan dari nilai *learning rate* dan jumlah *neuron hidden layer*. Hal ini dipengaruhi oleh nilai bobot yang diambil secara acak oleh sistem. Sehingga akan menghasilkan akurasi yang berbeda tiap pelatihan.

4.4. Hasil Klasifikasi menggunakan arsitektur jaringan terbaik

Salah satu arsitektur jaringan terbaik akan digunakan untuk melakukan uji coba klasifikasi penyakit DBD menggunakan data uji yang telah disediakan sebelumnya sebanyak 36 data. Tabel 5 akan menampilkan data hasil diagnosa penyakit DBD menggunakan metode LVQ.

Tabel 5. Data hasil klasifikasi menggunakan arsitektur jaringan terbaik

No	Pasien (x)	Hasil Klasifikasi	Target Asli
1	Pasien 1	DBD	DBD
2	Pasien 2	DBD	DBD
3	Pasien 3	DBD	DBD
4	Pasien 4	DBD	DBD
5	Pasien 5	DBD	DBD
6	Pasien 6	DBD	DBD
7	Pasien 7	DBD	DBD
8	Pasien 8	DBD	DBD
9	Pasien 9	DBD	DBD
10	Pasien 10	DBD	DBD
11	Pasien 11	DBD	DBD
12	Pasien 12	DBD	DBD
13	Pasien 13	DBD	DBD
14	Pasien 14	DBD	DBD
15	Pasien 15	DBD	DBD
16	Pasien 16	DBD	DBD
17	Pasien 17	DBD	DBD
18	Pasien 18	DBD	DBD
19	Pasien 19	DBD	DBD
20	Pasien 20	DBD	DBD
21	Pasien 21	Typhoid	Typhoid
22	Pasien 22	Typhoid	Typhoid
23	Pasien 23	Typhoid	Typhoid
24	Pasien 24	Typhoid	Typhoid
25	Pasien 25	Typhoid	Typhoid
26	Pasien 26	Typhoid	Typhoid
27	Pasien 27	Typhoid	Typhoid
28	Pasien 28	Typhoid	Typhoid
29	Pasien 29	DBD	Typhoid
30	Pasien 30	Typhoid	Typhoid
31	Pasien 31	Typhoid	Typhoid
32	Pasien 32	Typhoid	Typhoid
33	Pasien 33	Typhoid	Typhoid
34	Pasien 34	Typhoid	Typhoid
35	Pasien 35	Typhoid	Typhoid
36	Pasien 36	Typhoid	Typhoid

Proses pengujian menggunakan 36 data pasien yang terdiri dari 20 pasien dengan diagnosa penyakit DBD dan 16 pasien dengan diagnosa penyakit Typhoid. Hasil klasifikasi terdapat 35 data terklasifikasi benar serta 1 data terklasifikasi salah.

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan maka diperoleh kesimpulan bahwa algoritma LVQ dapat memahami pola dari proses pelatihan dan mampu mengklasifikasi penyakit DBD dengan nilai akurasi tertinggi 97.22%, dan nilai *Mean Square Error* (MSE) sebesar 0.028571. Nilai akurasi tersebut didapat dengan beberapa kombinasi parameter pelatihan yang

menghasilkan nilai akurasi yang sama. Salah satunya yaitu dengan kombinasi 0.1 *learning rate* dengan 10 *neuron hidden layer*. menghasilkan 1 data yang tidak berhasil di klasifikasi dengan benar dan 34 data di klasifikasi dengan benar.

5.2. Saran

Saran yang dapat diberikan oleh penulis untuk pengembangan berikutnya adalah sebagai berikut:

1. Untuk penelitian selanjutnya dapat mencoba variabel lain yang lebih mudah diperoleh seperti gejala umum yang bisa diketahui langsung oleh pasien.
2. Mencoba metode lain sebagai bahan perbandingan seperti menggunakan *Backpropagation* atau LVQ versi yang lebih baru.

DAFTAR PUSTAKA

Ahanger T.A. (2017). An Effective Approach of Detecting DDoS Using Artificial Neural Networks. International Conference on Wireless Comunications, Signal Processing and Networking. DOI: 10.1109/WiSPNET.2017.8299853

Amezcua J., Melin P., Castillo O. (2015). Design of an Optimal Modular LVQ Network for Classification of Arrhythmias Based on a Variable Training-Test Datasets Strategy. Springer International Publishing. DOI: 10.1007/978-3-319-11310-4_32

Budianita E., Novriyanto. (2015). Klasifikasi Status Gizi Balita Berdasarkan Indikator Antropometri Berat Badan Menurut Umur Menggunakan Learning Vector Quantization. SNTIKI. Vol.7. November 2015. ISSN :2085-9902

Devi K.J., Sravanthi K., Moulika G.B., Kumar K.M. (2017). Prediction of Medicines using LVQ Methodology. International Conference on Energy, Communication, Data Analytics and Soft Computing. DOI: 10.1109/ICECDS.2017.8390162

Felice D.D., Camastra F. (2018). Depth-Based Hand Pose Recognizer Using Learning Vector Quantization. Springer International Publishing. DOI: 10.1007/978-3-319-56904-8_7

Ghanem M., Mouloudi A., Mourchid M. (2016). Classification of Hadiths using LVQ based on VSM Considering Words Order. International Journal of Computer Applications. Vol.148. No.4. DOI: 10.5120/ijca2016911077

Leleury Z.A, Lesnussa Y.A., Madiuw J. (2016). Sistem Diagnosa Penyakit Dalam dengan Menggunakan Jaringan Saraf Tiruan Metode Backpropagation dan Learning Vector Quantization. Jurnal Matematika Interaktif Vol.12. No.2. DOI: 10.24198/jmi.v12.n2.11925.89-98

Rosario L.A.E., Duncan A.P., Lazaro P.A.M., Rejon J.E.G., Carro S.G., Ale J.F., Savic D.A., dan Karger F.E.M. (2018). Application of Artificial Neural Networks for Dengue Fever Outbreak Predictions in the Northwest Coast of Yucatan, Mexico and San Juan, Puerto Rico, Tropical Medicine and Infectious Disease. Vol.3. No.5. DOI: 10.3390/tropicalmed3010005

Setyawati O., Arifianto A.S., Sarosa M. (2017). Feature Selection for The Classification of Clinical Data of Stroke Patients. International Conference on Electrical Machines and Systems. DOI: 10.1109/ICEMS.2017.8056491

Shuo D., Xiao-heng C., Qing-hui W. (2014). A Study on the Application of Learning Vector Quantization Neural Network in Pattern Classification. Applied Mechanics and Materials. Vol.525. Hal.657-660. DOI: 10.4028/www.scientific.net/AMM.525.657

Sintawati I. D. (2016). Diagnosa penyakit dbd (demam berdarah dengue) dengan Algoritma pembelajaran hybrid dan backpropagation Berbasis neural network. Jurnal Sibernetika Vol.1. No.1. April 2016

Ying Z., Mei L. (2017). An Evaluation Model of Water Quality Based on Learning Vector Quantization Neural Network. Chinese Control Conference. DOI: 10.1109/ChiCC.2016.7553926

9 772527 583007