



Dewan Editor (*Editorial Board*)

Ketua Editor (*Chief Editor*)

Husni Thamrin

Editor Pelaksana (*Managing Editors*)

Fajar Suryawan, Universitas Muhammadiyah Surakarta
Gunawan Ariyanto, Universitas Muhammadiyah Surakarta
Nurgie Nurgiyatna, Universitas Muhammadiyah Surakarta

Anggota Dewan Editor (*Board of Editors*)

Teguh Bharata Adji, Universitas Gadjah Mada
Fajar Suryawan, Universitas Muhammadiyah Surakarta
Gunawan Ariyanto, Universitas Muhammadiyah Surakarta
Sri Karnila, Informatics and Business Institute Darmajaya, Bandar Lampung
Heru Supriyono, Universitas Muhammadiyah Surakarta
Nurgie Nurgiyatna, Universitas Muhammadiyah Surakarta

Mitra Bestari (*Reviewers*)

Mitra bestari yang terlibat dalam tiap penerbitan berbeda-beda. Daftar dapat dilihat pada sampul dalam bagian belakang.

Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika menerbitkan artikel ilmiah hasil riset atau review dalam bidang Ilmu Komputer atau Informatika secara umum yang meliputi bidang rekayasa perangkat lunak, pengembangan sistem informasi, sistem komputer, dan jaringan komputer. Jurnal ini diterbitkan oleh Muhammadiyah University Press, Universitas Muhammadiyah Surakarta.

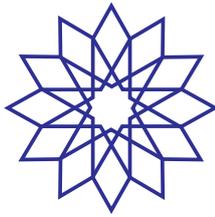
Khazanah Informatika: *Jurnal Ilmu Komputer dan Informatika* is a scientific journal that publishes scientific research papers/articles or reviews in the field of Computer Systems and Informatics. The scope of this journal includes software engineering, information systems development, computer systems and computer networking. *Khazanah Informatika* is published by Muhammadiyah University Press (MUP), Universitas Muhammadiyah Surakarta.



Crossref

DOAJ
DIRECTORY OF
OPEN ACCESS
JOURNALS





Pengantar Editor

Puji syukur kami haturkan ke hadirat Allah SWT atas segala nikmat dan izinnya sehingga jurnal Khazanah Informatika volume 3 nomor 1 bulan Juni 2017 dapat diterbitkan. Semoga penerbitan ini menambah koleksi dan direktori ilmu pengetahuan khususnya pada bidang ilmu komputer dan informatika.

Terbitan kali ini berisi tujuh artikel. Tiga artikel ditulis oleh penulis dari Universitas Muhammadiyah Surakarta, dan empat artikel lain ditulis oleh penulis dari Universitas Diponegoro, Universitas Muhammadiyah Magelang, Universitas Widya Dharma, dan Universitas PGRI Madiun. Dua artikel menyajikan hasil penelitian tentang sistem informasi, yaitu mencermati penerapan COBIT untuk penilaian tingkat kematangan sistem informasi dan tentang penerapan *e-commerce*. Dua artikel bertemakan penerapan ilmu komputer, yaitu jaringan syaraf tiruan untuk klasifikasi tipe kepribadian dan penerapan metode *Dempster-Shafer* untuk diagnosis gangguan telekomunikasi. Dua artikel bertemakan sistem komputer, yaitu aplikasi untuk deteksi kualitas toilet dan aplikasi untuk sistem keamanan sepeda motor. Akhirnya, satu artikel membahas pengujian mesin Nginx untuk *load balancing*.

Kami selaku editor mengucapkan terima kasih atas kepercayaan para penulis untuk mempublikasikan artikelnya di jurnal Khazanah Informatika. Kepercayaan ini akan kami jaga dengan merawat mutu jurnal ini. Selama bulan Januari-Juni 2017, kami mendapat kiriman 21 artikel, 7 artikel diputuskan untuk diterbitkan, 5 masih dalam proses *review* dan 9 artikel tidak diteruskan ke penerbitan. Mutu jurnal ini telah diakui oleh lembaga indeksal internasional karena sejak Oktober 2016, jurnal telah terindeks oleh lembaga bereputasi yaitu DOAJ (*Directory of Open Access Journal*).

Salam hangat

Chief Editor

Daftar Isi

Sistem Klasifikasi Tipe Kepribadian dan Penerimaan Teman Sebaya Menggunakan Jaringan Syaraf Tiruan <i>Backpropagation</i>	
<i>Yusuf Dwi Santoso, Subartono</i>	1-5
Pemanfaatan Web <i>E-Commerce</i> untuk Meningkatkan Strategi Pemasaran	
<i>Hani Atun Mumtabana, Sekreningsih Nita, Adzinta Winerawan Tito</i>	6-15
Sistem Pakar dengan Metode <i>Dempster Shafer</i> untuk Diagnosis Gangguan Layanan Indihome di PT Telkom Magelang	
<i>Endang Lestari, Emilya Ully Artha</i>	16-24
Identifikasi Kadar Amoniak sebagai Indikator Bau Toilet Menggunakan Perangkat MAS TUQUL	
<i>Salsabila Nurulfarah Mahmudah, Heni Hanifah, Wahyu Titis Satria Utama, Ismi Kamelia Najib Putri, Rahmat Kurniawan, Husni Thamrin</i>	25-30
Sistem Keamanan Ganda pada Sepeda Motor untuk Pencegahan Pencurian dengan Smarty (<i>Smart Security</i>)	
<i>Dendy Pratama, Eko Didik Febriyanto, Denisson Arif Hakim, Tri Muhyadi, Umi Fadlilah</i>	31-37
Penerapan Framework Cobit untuk Identifikasi Tingkat Kematangan Tata Kelola Teknologi Informasi: Studi Kasus di FASILKOM UNWIDHA	
<i>Agustinus Suradi, Sri Wiyanta</i>	38-42
Perancangan dan Pengujian <i>Load Balancing</i> dan <i>Failover</i> Menggunakan NGINX	
<i>Rahmad Dani, Fajar Suryawan</i>	43-50

Sistem Klasifikasi Tipe Kepribadian dan Penerimaan Teman Sebaya Menggunakan Jaringan Syaraf Tiruan *Backpropagation*

Yusuf Dwi Santoso^{1*}, Suhartono¹

¹Program Studi Ilmu Komputer/Informatika, Fakultas Sains dan Matematika
Universitas Diponegoro

*Yusufdwisantoso@gmail.com

Abstrak

Kepribadian merupakan gambaran tingkah laku dari individu. Penerimaan teman sebaya merupakan penilaian individu bahwa dirinya diterima, didengar, diperhatikan, dihargai, serta dapat merasa aman dan nyaman saat bersama teman-teman dengan umur yang sama. Kepribadian dan penerimaan teman sebaya penting untuk diketahui agar dapat mengenal potensi diri. Tes kepribadian merupakan salah satu sarana untuk mengetahui dan mengklasifikasikan kepribadian seseorang ke tipe kepribadian tertentu. Jaringan syaraf tiruan *backpropagation* dapat digunakan untuk melakukan klasifikasi sebuah pola berdasarkan permasalahan tertentu seperti halnya dalam mengklasifikasi tipe kepribadian dan penerimaan teman sebaya seseorang. Sistem klasifikasi tipe kepribadian dan penerimaan teman sebaya menggunakan jaringan syaraf tiruan *backpropagation* dapat digunakan untuk mengklasifikasi tipe kepribadian dan penerimaan teman sebaya seseorang ke dalam beberapa tipe, yaitu *introvert* diterima, *introvert* ditolak, *ekstrovert* diterima, dan *ekstrovert* ditolak berdasarkan sejumlah set pertanyaan yang menjadi alat ukur dalam penentuan kepribadian. Sistem klasifikasi tipe kepribadian dan penerimaan teman sebaya menggunakan jaringan syaraf tiruan *backpropagation* menghasilkan arsitektur *backpropagation* terbaik untuk klasifikasi kepribadian dan penerimaan teman sebaya pada saat menggunakan 1 *hidden layer* dengan 7 neuron, 10.000 *epoch*, nilai target *error* 0.01, dan laju pembelajaran 0.1. Hasil eksperimen jaringan syaraf tiruan *backpropagation* pada sistem ini menghasilkan rata-rata tingkat akurasi 98.75% dan tingkat *error* 1.25%.

Kata Kunci: klasifikasi; kepribadian; JST, *backpropagation*

1. Pendahuluan

Remaja adalah sosok yang senantiasa menarik untuk dibicarakan. Ketika seseorang memasuki masa remaja, maka saat itulah ia meninggalkan status dari anak-anak menuju masa dewasa dan ini yang sering disebut sebagai masa peralihan. Ciri-ciri yang menonjol pada usia remaja terlihat dalam perilaku sosialnya [1]. Bagi remaja, penerimaan sosial mempunyai arti penting yang mana salah satunya adalah diterima oleh teman sebaya. Tanpa penerimaan teman sekelompok akan menimbulkan gangguan perkembangan psikis dan sosial remaja. Terdapat beberapa faktor yang mempengaruhi penerimaan teman sebaya, salah satunya yaitu perilaku sosial yang ditampakkan oleh remaja [2].

Perilaku yang ditampakkan individu, tercermin dari gambaran orientasi terhadap dunianya. Orientasi ini disebut dengan sikap, di mana sikap dipengaruhi oleh tipe kepribadian individu. Kepribadian individu, secara sederhana dibedakan menjadi dua kutub berdasarkan arah energi psikis dalam diri individu, yaitu kepribadian *ekstrovert* dan *introvert* [3]. Kedua kepribadian ini memiliki caranya masing-masing dalam merespons lingkungannya, sehingga diperlukan sebuah tes yang berguna untuk mengenal kepribadian dan potensi diri yang ada agar dapat dioptimalkan untuk kesuksesan dalam kehidupan sosial.

Tes kepribadian adalah sebuah tes yang dilakukan untuk menentukan tipe kepribadian seseorang [4]. Pengklasifikasian tipe kepribadian yang dilakukan dengan tes kepribadian saat ini sering kali masih menggunakan metode lama dan kurang memanfaatkan teknologi yang ada. Padahal pesatnya perkembangan teknologi memberikan manfaat dalam kehidupan manusia. Pemanfaatan teknologi yang begitu besar membuat komputer dapat bekerja dengan meniru cara kerja otak manusia dengan memanfaatkan metode jaringan syaraf tiruan. Metode tersebut dapat dimanfaatkan untuk mengklasifikasi sebuah pola berdasarkan permasalahan tertentu seperti halnya dalam mengklasifikasikan tipe kepribadian dan penerimaan teman sebaya seseorang. Sehubungan dengan pentingnya pengenalan tipe kepribadian dan penerimaan teman sebaya serta dilatarbelakangi beberapa alasan tersebut maka perlu dibangunnya sebuah sistem klasifikasi tipe kepribadian dan penerimaan teman sebaya yang memanfaatkan teknologi komputer.

Klasifikasi kepribadian telah banyak diaplikasikan pada berbagai jurnal ilmiah. Menurut Lukas, Aribowo, dan Wardhani (2004) proses klasifikasi kepribadian yang menggunakan sistem pakar dengan metode ketidakpastian (*uncertainty*) ternyata hanya mampu menjawab 10 pertanyaan dari 20 pertanyaan yang ada dengan tingkat kepercayaan yang baik [5]. Klasifikasi

kepribadian menggunakan jaringan syaraf tiruan *learning vector quantization* memberikan nilai akurasi sebesar 63.75% [6]. Dari beberapa penelitian tersebut menunjukkan bahwa metode-metode yang telah digunakan sebelumnya belum memberikan hasil yang maksimal, untuk itu perlu dilakukan penelitian mengenai klasifikasi tipe kepribadian dan penerimaan teman sebaya dengan menggunakan algoritma *backpropagation*.

Salah satu model jaringan dalam jaringan syaraf tiruan adalah *backpropagation*. *Backpropagation* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respons yang benar terhadap pola masukan yang serupa dengan pola yang dipakai selama pelatihan [6]. Sudah banyak penelitian implementasi jaringan syaraf tiruan *backpropagation* terhadap prediksi seperti prediksi penyakit demam berdarah dengan akurasi 74% [7] dan kanker serviks dengan akurasi 95.14% [8].

Beberapa hasil penelitian mengenai algoritma *backpropagation* sebelumnya menunjukkan bahwa algoritma *backpropagation* mampu memberikan kinerja yang baik sehingga penulis tertarik untuk membuat sistem klasifikasi tipe kepribadian dengan menggunakan algoritma *backpropagation*. *Backpropagation* dapat digunakan untuk klasifikasi karena *backpropagation* dapat mengenali pola dan memberikan respons benar terhadap pola masukan sehingga cocok untuk mengklasifikasikan kepribadian seseorang berdasarkan input pertanyaan. Sistem klasifikasi kepribadian yang memanfaatkan algoritma *backpropagation* ini diharapkan mampu memberikan hasil *output* yang lebih akurat. Sistem klasifikasi kepribadian dan penerimaan teman sebaya ini dibangun berbasis web. Kelebihan pembangunan sistem berbasis web adalah dapat diakses dengan mudah dari jarak jauh melalui *browser* tanpa harus melakukan instalasi *software* sehingga nantinya sistem klasifikasi tipe kepribadian dan penerimaan teman sebaya ini dapat digunakan oleh kalangan umum di manapun dan kapanpun tanpa bantuan seorang pakar secara langsung.

Tujuan yang ingin dicapai adalah terciptanya suatu sistem klasifikasi tipe kepribadian dan penerimaan teman sebaya menggunakan jaringan syaraf tiruan *backpropagation*, sehingga dapat memudahkan seseorang untuk mengenali kepribadiannya masing-masing.

2. Metode

2.1 Pengumpulan Data

Pada penelitian ini, data diperoleh melalui kuesioner. Kuesioner disebar kepada mahasiswa tingkat satu sampai dengan empat dengan rentang usia 18 sampai 22 tahun melalui *google form*. Jumlah data yang telah terkumpul sebanyak 210 data kemudian dikelompokkan ke dalam 4 kelas melalui bantuan psikolog, sehingga data sebanyak 210 buah ini terdiri atas 69 data yang didefinisikan ke dalam kelas *ekstrovert* diterima, 49 data untuk *introvert* diterima, 52 data untuk *ekstrovert* ditolak, 40 data untuk *introvert* ditolak.

2.2 Arsitektur JST *Backpropagation*

Arsitektur JST *backpropagation* yang digunakan pada sistem ini terdiri atas 3 *layer*, yaitu *input layer*, *hidden layer*, dan *output layer*.

2.3 Jaringan Syaraf Tiruan *Backpropagation*

Jaringan syaraf tiruan adalah sistem pemroses

informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi. Jaringan syaraf tiruan ditentukan oleh 3 hal, yaitu arsitektur jaringan, metode untuk menentukan bobot penghubung, serta fungsi aktivasi [4].

Backpropagation merupakan salah satu model jaringan syaraf tiruan. *Backpropagation* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respons yang benar terhadap pola masukan yang serupa tapi tidak sama dengan pola yang dipakai selama pelatihan [8].

Arsitektur *backpropagation* terdiri atas satu atau lebih unit masukan ditambah satu unit bias, satu layer tersembunyi yang terdiri atas satu atau lebih unit ditambah satu unit bias, serta satu atau lebih unit keluaran.

Algoritma pelatihan *backpropagation* adalah sebagai berikut:

- Langkah 0: Inisialisasi semua bobot dengan bilangan acak kecil, $epoch = 1$, tentukan laju pembelajaran (α), tentukan jumlah unit pada layer tersembunyi (p) serta tentukan kondisi penghentian. Kondisi penghentian berupa maksimum $max\ epoch$ dan $error\ target$.
 - Langkah 1: Jika $epoch \neq max\ epoch$ dan $error\ target < MSE$, lakukan langkah 2–9.
 - Langkah 2: Untuk setiap pasang data pelatihan (1 sampai dengan a di mana a adalah jumlah data pelatihan), lakukan langkah 3–8.
 - Fase I: Propagasi maju
- Langkah 3 : Tiap unit masukan menerima sinyal dan meneruskannya ke unit tersembunyi di atasnya.
- Langkah 4 : Hitung semua keluaran di unit tersembunyi z_j ($j = 1, 2, \dots, p$)

$$z_{-net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ji} \quad (1)$$

$$z_j = f(z_{-net_j}) = \frac{1}{1 + e^{-z_{-net_j}}} \quad (2)$$

Langkah 5: Hitung semua keluaran jaringan di unit y_k ($k = 1, 2, \dots, m$)

$$y_{-net_k} = w_{k0} + \sum_{j=1}^p z_j w_{kj} \quad (3)$$

$$y_k = f(y_{-net_k}) = \frac{1}{1 + e^{-y_{-net_k}}} \quad (4)$$

e. Fase II : Propagasi mundur

Langkah 6: Hitung faktor δ unit keluaran berdasarkan kesalahan di setiap unit keluaran y_k ($k = 1, 2, \dots, m$)

$$\delta_k = (t_k - y_k) f'(y_{-net_k}) = (t_k - y_k) y_k (1 - y_k) \quad (5)$$

Hitung suku perubahan bobot W_{kj} dengan laju percepatan α

$$\Delta w_{kj} = \alpha \delta_k z_j \quad (6)$$

$$k = 1, 2, \dots, m; j = 0, 1, \dots, p$$

Langkah 7: Hitung faktor δ unit tersembunyi berdasarkan kesalahan di tiap unit tersembunyi z_j ($j = 1, 2, \dots, p$)

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj} \quad (7)$$

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} z_j (1 - z_j) \quad (8)$$

Hitung suku perubahan bobot V_j dengan laju percepatan α

$$\Delta v_{ji} = \alpha \delta_j x_i \quad (9)$$

$$j = 1, 2, \dots, p; i = 0, 1, \dots, n$$

f. Fase III : Perubahan bobot

Langkah 8: Hitung semua perubahan bobot

Perubahan bobot garis yang menuju ke unit keluaran

$$w_{kj}(\text{baru}) = w_{kj}(\text{sekarang}) + \Delta w_{kj} \quad (10)$$

$$k = 1, 2, \dots, m; j = 0, 1, \dots, p$$

Perubahan bobot garis yang menuju ke unit tersembunyi:

$$v_{kj}(\text{baru}) = v_{kj}(\text{sekarang}) + \Delta v_{ji} \quad (11)$$

$$j = 1, 2, \dots, p; i = 0, 1, \dots, n$$

g. Langkah 9: Perbaharui nilai *epoch*

$$\text{epoch} = \text{epoch} + 1 \quad (12)$$

dan hitung *mean squared error (MSE)*

$$MSE = \frac{\sum_{k=1}^a (t_k - y_k)^2}{a} \quad (13)$$

2.4 Algoritma Nguyen Widrow

Bobot awal akan mempengaruhi apakah jaringan mencapai titik minimum lokal atau global dan seberapa cepat konvergensinya. Nguyen dan Widrow (2010) mengusulkan cara membuat inisialisasi bobot ke unit tersembunyi sehingga menghasilkan iterasi yang lebih cepat. Algoritma Nguyen Widrow adalah sebagai berikut [9]:

1. Inisialisasi semua bobot (V_{ji} (Lama)) dengan bilangan acak dalam interval $[-0.5, 0.5]$
2. Hitung:

$$\|v_{ji}\| = \sqrt{v_{j1}^2 + v_{j2}^2 + \dots + v_{jn}^2} \quad (14)$$

3. Bobot yang digunakan sebagai inisialisasi:

$$v_{ji} = \frac{\beta v_{ji}(\text{lama})}{\|v_j\|} \quad (15)$$

Di mana β adalah faktor skala dengan rumus:

$$\beta = 0.7 \sqrt{p} \quad (16)$$

4. Bobot bias yang dipakai sebagai inisialisasi (V_{j0}) adalah bilangan acak dalam interval $[-\beta, \beta]$

2.5 Preprocessing Data

Preprocessing data merupakan suatu proses atau langkah yang dilakukan untuk membuat data mentah (dalam penelitian ini berupa data hasil kuesioner) menjadi data yang siap diolah menggunakan metode *backpropagation*. Pada penelitian ini *preprocessing* data dilakukan dalam 3 tahapan, yaitu *mapping* data, normalisasi data, dan pemisahan antara data latih dan data uji dengan metode *K-Fold*.

Sistem klasifikasi kepribadian ini akan menggunakan sejumlah 10 atribut pertanyaan untuk menentukan tipe kepribadian dan penerimaan teman sebaya seseorang. Setiap hasil tes yang berjumlah 10 atribut pertanyaan ini akan direpresentasikan sebagai variabel input, sedangkan setiap atribut pertanyaan akan direpresentasikan sebagai variabel input. Dari masing-masing atribut pertanyaan tersebut terdapat 4 pilihan jawaban di mana akan merepresentasikan karakteristik dari masing-masing tipe kepribadian.

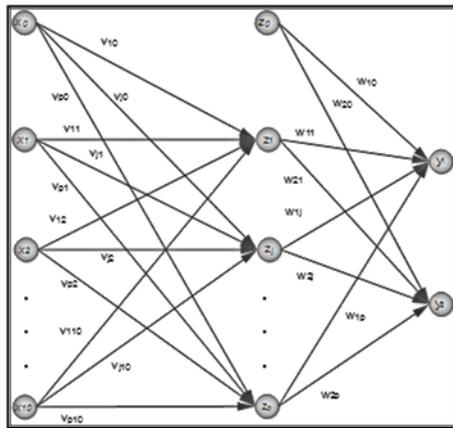
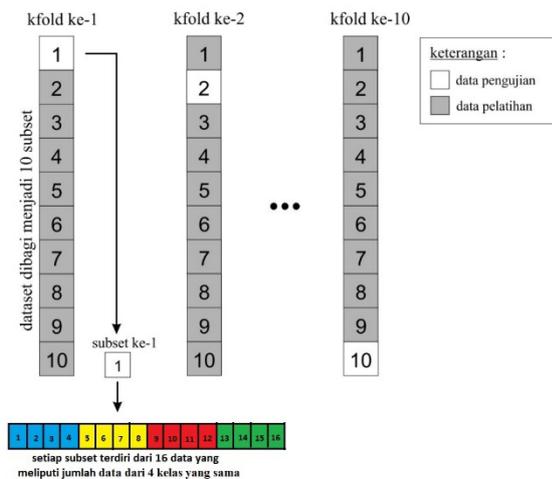
Langkah *mapping* data dilakukan untuk mengidentifikasi data yang telah diperoleh dalam pengumpulan data ke dalam arsitektur *backpropagation*. Data yang diperoleh diidentifikasi dan selanjutnya dipetakan data mana yang akan dijadikan *input neuron* dan data mana yang akan menjadi *output neuron*.

Langkah kedua yaitu *mapping* data *input* dan *output* jaringan. Pertanyaan dari kuesioner akan menjadi *input neuron*, pertanyaan ke-1 akan menjadi x_1 , pertanyaan ke-2 akan menjadi x_2 , begitu seterusnya sampai dengan karakteristik layanan ke 10 akan menjadi x_{10} . Sedangkan nilai *output neuron* diperoleh dari hasil perhitungan sehingga terbentuk arsitektur jaringan dengan 10 neuron *input* dan 1 neuron *output* seperti pada Gambar 1.

Langkah ketiga yaitu normalisasi data yang telah dipetakan ke dalam arsitektur jaringan dengan rumus berikut:

$$x = \frac{r - 1}{R - 1} \quad (17)$$

Langkah 4, yaitu identifikasi data latih dan data uji menggunakan *K-Fold Cross Validation* dengan nilai $K=10$. Data yang telah dilakukan *preprocessing* kemudian akan diproses menggunakan *K-Fold Cross Validation*. Tahapan ini dilakukan untuk membagi data hasil menjadi data pelatihan dan data pengujian. Penelitian ini menggunakan *dataset* sejumlah 160 yang terdiri atas 4 kelas tipe kepribadian.

Gambar 1. Arsitektur JST *Backpropagation*

Gambar 2. 10-Fold Cross Validation pada Dataset

Proses *K-Fold Cross Validation* ini menggunakan nilai, sehingga dataset akan dibagi menjadi 10 *subset* yang terdiri atas data dengan perbandingan jumlah kelas yang sama. Setiap *subset* akan terdiri 16 data, di mana 16 data tersebut terdiri atas 4 kelas yang masing-masing terdapat 4 data. Metode ini akan melakukan iterasi sebanyak 10 kali, di mana pada iterasi pertama subset ke-1 menjadi data pengujian, sedangkan sisa subset lainnya akan menjadi data pelatihan. Pada iterasi kedua, subset kedua digunakan sebagai data pengujian dan subset lainnya sebagai data pelatihan, dan seterusnya hingga seluruh *subset* digunakan sebagai data pelatihan. Penggambaran metode 10-Fold *Cross Validation* pada dataset lebih jelasnya ditunjukkan pada Gambar 2.

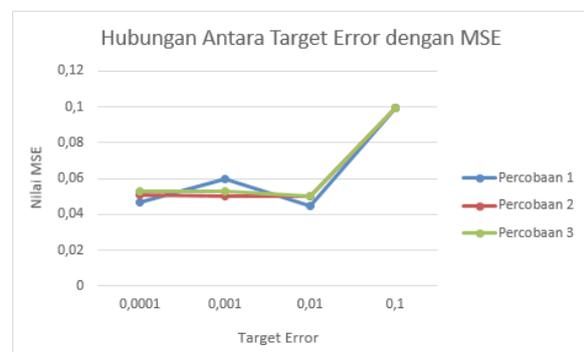
Dataset awal sejumlah 160 dengan proses 10-Fold *Cross Validation* ini akan diperoleh data pelatihan sebanyak 144 data dan data pengujian sebanyak 16 data dalam setiap pelatihan.

3. Hasil

Hasil penelitian berdasar eksperimen nilai target *error*, eksperimen kombinasi *hidden neuron*, dan laju pembelajaran.

Tabel 1. Hasil Percobaan Target *Error*

Percobaan ke-	Target <i>Error</i>	Rata-rata MSE
1	0.1	0.09940154
	0.01	0.04467202
	0.001	0.05989982
	0.0001	0.04645378
2	0.1	0.099472963
	0.01	0.055573744
	0.001	0.050438866
	0.0001	0.050739845
3	0.1	0.099216491
	0.01	0.050032963
	0.001	0.052557226
	0.0001	0.052302219

Gambar 3. Grafik Hubungan Target *Error* dengan MSE

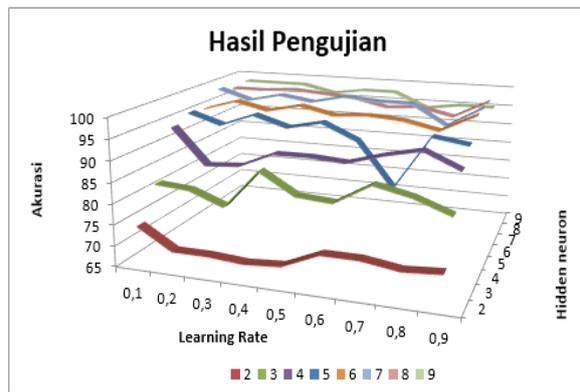
3.1 Eksperimen Nilai Target *Error*

Pengujian target *error* dilakukan dengan menggunakan nilai target *error* adalah 0.1; 0.01; 0.001; 0.0001. Variabel lain yang digunakan pada pengujian ini yaitu jumlah *hidden neuron* sebanyak 3, laju pembelajaran (α) sebesar 0.4, dan *epoch* maksimum sebanyak 100.000 *epoch*. Hasil dari pengujian target *error* dalam 3 kali percobaan ditunjukkan pada Tabel 1.

Nilai rata-rata *mean square error* terendah pada ketiga percobaan selalu pada nilai target *error* 0.01. Hasil eksperimen pengujian ke-1 ini kemudian disajikan dalam grafik yang dapat dilihat pada Gambar 3

3.2 Eksperimen Kombinasi *Hidden Neuron* dan Laju Pembelajaran

Nilai laju pembelajaran yang diteliti antara 0.1; 0.2; 0.3; 0.4; 0.5; 0.6; 0.7; 0.8; 0.9. Jumlah *hidden neuron* yang diteliti 2 sampai dengan 9. Nilai maksimum *epoch* yang digunakan adalah 10.000 dan nilai target *error* yang digunakan sesuai hasil dari strategi 1 yaitu 0.01. Sehingga akan terbentuk 72 kombinasi parameter penelitian. Masing-masing kombinasi tersebut akan menghasilkan kombinasi jaringan terbaik dari pembangunan sistem ini dengan nilai *error* minimum 0.01 dan *epoch* maksimal 10.000.



Gambar 4. Grafik Hasil Pengujian

Arsitektur terbaik dari jaringan syaraf tiruan *backpropagation* didapatkan dari kombinasi laju perubahan dan jumlah *hidden neuron*. Sesuai dengan hasil penelitian di atas, kombinasi *hidden neuron* dan *alfa* yang memiliki tingkat keakuratan tertinggi yaitu pada nilai *hidden neuron* 7 dan dengan laju percepatan 0,5 dengan tingkat akurasi 98,75%. Hasil eksperimen pengujian ke-2 ini kemudian disajikan dalam grafik yang dilihat pada Gambar 4.

4. Diskusi

Klasifikasi kepribadian telah banyak dipublikasikan pada berbagai jurnal ilmiah. Pada beberapa penelitian terdahulu seperti Lukas, Aribowo, dan Wardhani disebutkan bahwa proses klasifikasi kepribadian yang menggunakan sistem pakar dengan metode ketidakpastian (*uncertainty*) ternyata hanya mampu menjawab 10 pertanyaan dari 20 pertanyaan yang ada dengan tingkat kepercayaan yang baik [5]. Klasifikasi kepribadian menggunakan jaringan syaraf tiruan *learning vector quantization* memberikan nilai akurasi sebesar 63.75% [6]. Sudah banyak penelitian implementasi jaringan syaraf tiruan *backpropagation* terhadap prediksi seperti prediksi penyakit demam berdarah dengan akurasi 74% [7] dan kanker serviks dengan akurasi 95.14% [8]. Dari beberapa penelitian tersebut menunjukkan bahwa metode-metode yang telah digunakan sebelumnya belum memberikan hasil yang maksimal. Sedangkan penelitian mengenai klasifikasi tipe kepribadian dan penerimaan teman sebaya dengan menggunakan algoritma *backpropagation*. menghasilkan tingkat akurasi 98.75%.

5. Kesimpulan

Sistem klasifikasi kepribadian dan penerimaan teman sebaya menggunakan jaringan syaraf tiruan *backpropagation* telah berhasil dibangun dan dapat berjalan sesuai kebutuhan fungsional. Sistem klasifikasi kepribadian dan penerimaan teman sebaya menggunakan jaringan syaraf tiruan *backpropagation* menghasilkan arsitektur jaringan terbaik dengan 10 neuron pada *input layer*, 7 neuron pada *hidden layer*, dan 2 neuron *output* mampu melakukan klasifikasi dengan tingkat akurasi terbaik, yaitu sebesar 98.75% dan tingkat *error* 1.25%.

6. Daftar Pustaka

- [1] I. Puhar, "Penerimaan Teman Sebaya Pada Remaja Panti Asuhan Ditinjau Dari Kematangan Sosial," Skripsi, Unika Soegijapranata, p. 63, 2007.
- [2] R. Sinthia, "Hubungan Antara Penerimaan Sosial Kelompok Kelas dengan Kepercayaan Diri Pada Siswa Kelas I SLTP XXX Jakarta," Triadik, vol. 14, no. 1, p. 42, 2011.
- [3] Suryabrata, Psikologi Kepribadian, Jakarta: Rajawali Pers, 2005.
- [4] F. Littauer, Personality Plus, Tangerang: Karisma Publishing Group, 2011.
- [5] S. Lukas, A. Ariwibowo dan R. Wardhani, "Sistem Pakar Untuk Mendiagnosa Kepribadian Berdasarkan Teori Kepribadian Florence Littauer," *Jurnal Ilmiah Ilmu Komputer*, p. 129, 2004.
- [6] A. M. Pratiwi dan P. S. Sasongko, "Sistem Klasifikasi Kepribadian Menggunakan Jaringan Syaraf Tiruan," *Journal of Informatics and Technology*, p. 8, 2015.
- [7] W. Widodo, A. Rachman dan R. Amelia, "Jaringan Syaraf Tiruan Prediksi Penyakit Demam Berdarah dengan Menggunakan Metode Backpropagation," *Jurnal IPTEK*, vol. 18, no.1, 2014.
- [8] A. T. Susanto, "Aplikasi Diagnosa Kanker Serviks dengan Menggunakan Algoritma Backpropagation," Skripsi, 2012.
- [9] J. Siang, Jaringan Syaraf Tiruan dan Pemrograman Menggunakan Matlab, Yogyakarta: Penerbit Andi, 2005.

Pemanfaatan Web *E-Commerce* untuk Meningkatkan Strategi Pemasaran

Hani Atun Mumtahana^{1*}, Sekreningsih Nita¹, Adzinta Winerawan Tito¹

^{1*}Program Studi Teknik Informatika

Universitas PGRI Madiun

*hany_alea03@yahoo.com

Abstrak

UMKM merupakan salah satu usaha mikro yang memberdayakan industri rumahan. UMKM Indonesia memiliki kontribusi sebesar 15.8% terhadap rantai pasok produksi global di tingkat ASEAN (1). Perkembangan UMKM dari tahun ke tahun dipengaruhi dengan pemanfaatan teknologi informasi dan sistem informasi. *E-commerce* merupakan salah satu teknologi yang mendukung perkembangan UMKM dan perdagangan saat ini. Pengembangan aplikasi *e-commerce* ini dibangun dengan menggunakan model pengembangan perangkat lunak *waterfall*. Tahapan pengembangan dimulai dengan analisis permasalahan dan kebutuhan untuk aplikasi pemasaran, desain model proses, desain model data, pembuatan program aplikasi, implementasi aplikasi, dan evaluasi aplikasi. Penelitian ini bertujuan untuk mengetahui dampak yang dirasakan atas pemanfaatan teknologi *e-commerce* dalam peningkatan strategi pemasaran produksi. Teknologi *e-commerce* dibangun dengan menggunakan aplikasi berbasis *website*, sehingga para pelaku usaha dapat mempromosikan hasil usaha dengan mudah. Sasaran utama pemanfaatan teknologi *e-commerce* ini adalah UMKM industri kerajinan kulit di Magetan. Pengguna dari aplikasi ini adalah administrator yang mempunyai hak penuh terhadap pengolahan aplikasi, operator atau *owner* merupakan pemilik toko yang memiliki hak untuk memasukkan data produk, pengguna yang memiliki hak untuk memilih dan memesan produk. Hasil uji coba dan evaluasi yang dilakukan pada koperasi mahasiswa STT Dharma Iswara Madiun, menunjukkan bahwa dengan memanfaatkan teknologi *e-commerce* memenuhi strategi pemasaran produk yang dapat berdampak pada peningkatan pendapatan.

Kata Kunci: UMKM, teknologi *e-commerce*, teknologi informasi dan sistem informasi, UMKM industri kerajinan kulit, koperasi mahasiswa STT Dharma Iswara Madiun

1. Pendahuluan

Perkembangan teknologi informasi saat ini merupakan suatu kebutuhan yang harus dimiliki suatu organisasi untuk meningkatkan kualitas layanan. Pemanfaatan teknologi informasi harus diimbangi dengan implementasi sistem informasi. Teknologi informasi dan sistem informasi merupakan dua bidang yang saling berdampingan dalam penyampaian informasi bagi suatu organisasi. Pemanfaatan teknologi informasi dan sistem informasi pada suatu organisasi dapat membantu organisasi untuk mencapai tujuan dengan maksimal. Hal ini terbukti dengan semakin berkembang dan mudahnya peran teknologi informasi dan sistem informasi di kalangan organisasi.

Usaha mikro berdasarkan undang-undang nomor 20 tahun 2008 tentang UMKM (Usaha Menengah Kecil dan Mikro) adalah usaha produktif milik orang perorangan dan/atau badan usaha perorangan yang memenuhi kriteria usaha mikro sebagaimana diatur dalam undang-undang.

Pada penelitian [1] menguraikan tentang penerapan *e-commerce* untuk pengembangan produksi UKM di Kabupaten Semarang sangat berpeluang besar. Sampel

penelitian pada 30 UKM yang ada di Kabupaten Semarang menunjukkan nilai skor 3,50 untuk persepsi pengguna terhadap manfaat *e-commerce*. Namun, persepsi pengguna terhadap kendala terhadap *e-commerce* juga sangat tinggi, yaitu skor rata-rata 3,48. Kendala tersebut muncul karena para pelaku bisnis UKM belum dapat sepenuhnya menggantikan cara tradisional dengan memanfaatkan teknologi informasi dalam melakukan transaksi.

E-commerce merupakan teknologi yang menjadi kebutuhan mendasar setiap organisasi yang bergerak di bidang perdagangan. *E-commerce* merupakan cara bagi konsumen untuk dapat membeli barang yang diinginkan dengan memanfaatkan teknologi internet [2]. Pemanfaatan teknologi *e-commerce* dapat dirasakan oleh konsumen (*business to consumer*) maupun oleh pelaku bisnis (*business to business*).

Pemanfaatan teknologi *e-commerce* oleh para pelaku bisnis dapat memberikan nilai positif maupun nilai negatif. Hasil penelitian [3] menjelaskan bahwa penelitian terhadap variabel perspektif risiko dan kepercayaan pelanggan terhadap *e-commerce* menunjukkan adanya proses saling mempengaruhi antara dua variabel tersebut.

Hal ini ditunjukkan dengan variabel persepsi konsumen terhadap risiko yang ditimbulkan *e-commerce* cukup tinggi. Sedangkan kepercayaan konsumen terhadap *e-commerce* cenderung rendah. Salah satu faktor pengaruh tersebut adalah masih bergantungnya cara pemasaran tradisional.

Perkembangan UMKM di Magetan yang sudah banyak dikenal adalah kerajinan industri kulit. Kerajinan yang berlokasi di Jalan Sawo Kabupaten Magetan ini sudah banyak menarik minat wisatawan domestik dan mancanegara. Pertumbuhan pengrajin kulit di Magetan mulai tahun 1960 hingga saat ini terus mengalami pertumbuhan yang signifikan. Sampai saat ini sudah terdapat 36 UKM yang menghasilkan kerajinan kulit.

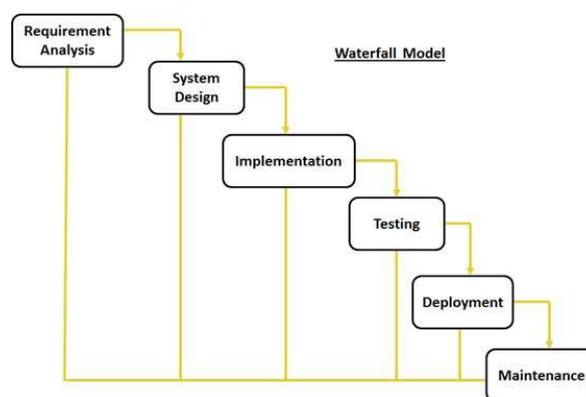
Permasalahan yang masih dirasakan oleh para pengrajin kulit di Kabupaten Magetan adalah kurangnya media yang digunakan untuk memasarkan hasil produksi mereka. Cara pemasaran tradisional masih menjadi andalan para pengrajin. Pusat perdagangan di Jalan Sawo Kabupaten Magetan merupakan salah satu lokasi yang paling banyak didatangi oleh para pelanggan. Selain menjajakan hasil kerajinan di setiap toko, para pengrajin juga mengikuti pameran yang diselenggarakan oleh daerah. Cara pemasaran yang masih bersifat tradisional ini sangat terbatas dalam penyampaian informasi hasil kerajinan kulit. Sehingga pemasaran hasil kerajinan yang sudah banyak dikenal oleh wisatawan domestik maupun mancanegara sangat kurang.

Pada penelitian [4] menghasilkan suatu strategi yang perlu digunakan untuk meningkatkan nilai pendapatan para pengrajin industri kulit di Kabupaten Magetan. Hasil penelitian menunjukkan berdasarkan hasil analisis lingkungan internal dan eksternal dengan menggunakan metode SWOT matrik dan PEST maka perlu adanya pemanfaatan teknologi informasi dan sistem informasi dalam bentuk *e-commerce* untuk meningkatkan kualitas pemasaran hasil produksi.

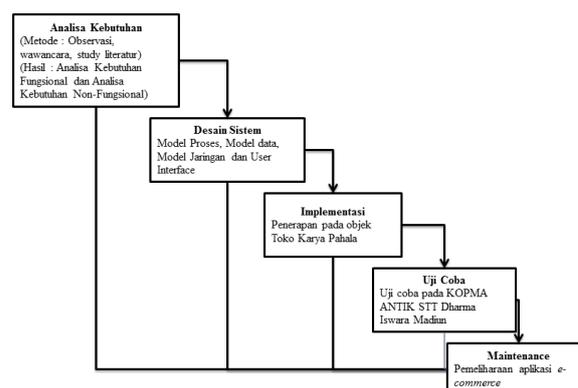
Penelitian ini akan menghasilkan sebuah aplikasi *e-commerce* yang dapat digunakan oleh pelaku bisnis sentra industri kulit Magetan. Pengembangan aplikasi dengan menggunakan model pengembangan perangkat lunak *waterfall* akan menghasilkan sebuah aplikasi yang sesuai dengan kebutuhan pengguna. Aplikasi ini dirancang dengan hak akses pengguna yaitu: administrator, pemilik toko (*owner*), dan konsumen. Aplikasi ini akan dibangun dengan menggunakan teknologi berbasis *website* sehingga setiap informasi terbaru akan mudah diterima oleh setiap pengguna.

Pada pengembangan aplikasi ini dirancang untuk sistem pemesanan dan pembelian secara *online*. Keamanan transaksi jual beli secara *online* merupakan salah satu variabel yang mempengaruhi minat konsumen akan menjadi pembahasan yang paling diperhatikan. Konsumen akan diberikan pilihan sistem pembayaran yaitu COD, sistem *paypal*, maupun transfer.

Berdasarkan permasalahan tersebut, penelitian ini akan mengembangkan sebuah aplikasi *e-commerce* sebagai salah satu strategi pemasaran untuk para pengrajin industri kulit di Kabupaten Magetan. Pengembangan aplikasi ini bertujuan untuk memudahkan para pelaku bisnis dalam memberi dan menerima informasi.



Gambar 1. Model pengembangan *Waterfall* [6]



Gambar 2. Tahapan penelitian

2. Metode

Metodologi penelitian merupakan jabanran langkah-langkah yang akan dilakukan dalam pelaksanaan penelitian. Adapun Metodologi yang akan digunakan dalam penelitian ini adalah metodologi kualitatif. Menurut [5] metodologi penelitian kualitatif adalah suatu metodologi yang bermaksud untuk memahami fenomena tentang apa yang dialami oleh subjek penelitian. Metodologi ini merupakan metodologi yang mengesampingkan model perhitungan matematika dan statistik.

Dalam melakukan pengembangan aplikasi *e-commerce* digunakan model pengembangan perangkat lunak *waterfall*. Model pengembangan perangkat lunak *waterfall* adalah [6] sebuah model pengembangan perangkat lunak yang menggunakan prinsip pengembangan secara berurutan. Tahapan selanjutnya tidak akan dikerjakan jika tahapan sebelumnya belum terselesaikan. Adapun gambaran tahapan pada model pengembangan *waterfall* adalah sebagai berikut:

Pengembangan aplikasi *e-commerce* ini melalui 5 tahapan sesuai dengan model pengembangan yang digunakan. Adapun rincian kegiatan yang dilakukan pada setiap tahapan adalah sebagai berikut:

2.1 Tahap 1: Analisis Kebutuhan

Pada tahap ini dilakukan analisis kebutuhan untuk pengembangan aplikasi *e-commerce*. Metode yang digunakan dalam pengumpulan data yaitu, wawancara, pengamatan

secara langsung, dan literatur pustaka. Pengamatan secara langsung dilakukan di sentra industri kulit Jalan Sawo Kabupaten Magetan khususnya toko Karya Pahala dan Barokah. Selain itu wawancara secara langsung dilakukan pada pemilik toko dengan kesimpulan hasil wawancara yaitu mengenai metode pemasaran yang dilakukan oleh para pengrajin yang sebagian besar masih menggunakan cara tradisional. Selain itu metode promosi juga dinaungi oleh paguyuban para pengrajin yang secara langsung berada di bawah pengawasan Dinas Pariwisata Kabupaten Magetan.

Setelah melakukan pengamatan dan wawancara secara langsung, pada tahap analisis juga dilakukan analisis kebutuhan fungsional dan kebutuhan non-fungsional pengembangan aplikasi *e-commerce*. Hasil analisis merupakan acuan dalam pembuatan desain model proses, model data, dan antarmuka pengguna.

2.2 Tahap 2: Desain Sistem

Pada tahap ini akan dibuat desain model proses, model data, dan desain antarmuka. Desain model proses digambarkan dengan *United Modelling Language (UML)*.

2.3 Tahap 3: Implementasi

Pada tahap implementasi yaitu penerapan aplikasi *e-commerce* sesuai dengan kebutuhan dari toko Karya Pahala yang merupakan salah satu sentra terbesar di Jalan Sawo.

2.4 Tahap 4: Uji Coba

Pada tahap uji coba, dilakukan uji coba kegunaan aplikasi *e-commerce* pada koperasi mahasiswa di STT Dharma Iswara Madiun. Hasil uji coba digunakan untuk menilai tingkat kegunaan aplikasi yang telah dikembangkan sebelum ada evaluasi untuk pengembangan aplikasi lebih lanjut.

2.5 Tahap 5: Maintenance

Pada tahap ini dilakukan pemeliharaan aplikasi *e-commerce*. Langkah pemeliharaan masih dilakukan sampai dengan akhir tahun 2016.

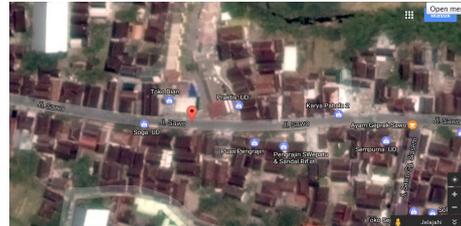
Pada proses pengembangan belum dilakukan tahapan implementasi pada objek penelitian. Hal ini dikarenakan tahap pengembangan hasil penelitian yang masih perlu adanya penyempurnaan. Hasil akhir penerapan aplikasi *e-commerce* pada Kopma Antik STT Dharma Iswara Madiun menyatakan perlu adanya evaluasi lebih lanjut terhadap aplikasi yang akan dikembangkan pada sentra industri kulit Kabupaten Magetan.

3. Hasil

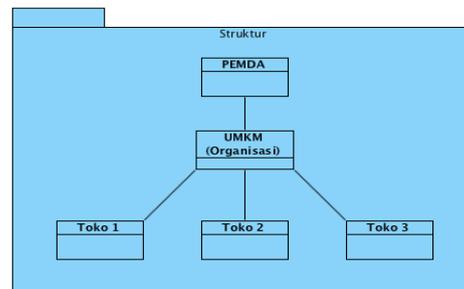
Pada penelitian ini menghasilkan sebuah rancang bangun aplikasi *e-commerce* untuk pemasaran produksi industri kulit di Kabupaten Magetan. Model pengembangan yang digunakan pada penelitian ini adalah model *waterfall* yang merupakan adopsi dari siklus hidup pengembangan perangkat lunak yang dimulai dari tahap analisis data dan kebutuhan sistem, desain sistem, pembuatan perangkat lunak, dan pengujian sistem.



Gambar 3. Gapura Sentra Industri Kulit Magetan



Gambar 4. Lokasi Sentra Industri Kulit Magetan



Gambar 5. Struktur Organisasi UMKM

Hasil dari penelitian yang merupakan sebuah teknologi terapan yang akan dipergunakan sebagai media promosi hasil industri kulit di Kabupaten Magetan, sehingga dapat memperluas pasar promosi yang sampai dengan saat ini telah merambah ke pasar nasional dan pasar internasional. Kelebihan dari aplikasi *e-commerce* ini terletak pada sistem pembayaran yang sudah menerapkan teknologi *paypal*. Pemilihan teknologi *paypal* sebagai salah satu sistem pembayaran dikarenakan sudah hampir 30% pemasaran hasil industri kulit terjadi di pasar internasional. Pada hasil penelitian yang dilakukan mulai bulan Februari 2016 sampai dengan Desember 2016 akan dijelaskan secara rinci. Adapun hasil tersebut adalah sebagai berikut:

3.1 ANALISIS SISTEM LAMA

Produksi kerajinan kulit Magetan memiliki tiga lokasi di antaranya, sentra satu di Kelurahan Selosari, sentra dua di Desa Kauman, dan sentra tiga di Desa Mojopurno Kabupaten Magetan. Sentra satu tepatnya berlokasi di Jalan Sawo merupakan wilayah yang paling terkenal, karena merupakan daerah strategis dan dilalui banyak wisatawan menuju telaga Sarangan.

Kerajinan kulit ini merupakan implementasi dari kegiatan penyamakan kulit yang merupakan kebudayaan Kabupaten Magetan. Kerajinan kulit yang telah dirintis sejak tahun 1960 hingga saat ini telah menghasilkan banyak produk. Produk yang paling terkenal yaitu alas kaki (sandal dan sepatu), tetapi ada beberapa kerajinan lain yang dihasilkan antara lain, ikat pinggang, tas, jaket, aksesoris.

Tabel 1. Analisis PIECES

Analisis PIECES	Sistem yang Sedang Berjalan
Kinerja (performance)	Sangat sulit melakukan pengecekan data produk yang ada dengan produk yang telah terjual karena media penyimpanannya melalui kertas.
Informasi (information)	Promosi yang dilakukan oleh UKM masih kurang sehingga masih banyak masyarakat yang tidak mengenal industri kulit Magetan.
Analisis ekonomi (economic)	Biaya yang tinggi untuk memasang iklan di media cetak, elektronik, dan percetakan brosur.
Keamanan atau control (security or control)	Tidak adanya perbedaan antara <i>user</i> biasa (pengunjung) dengan <i>user</i> anggota (pelanggan).
Efisiensi (efficiency)	Karyawan harus mencatat data yang memesan produk via telepon dan sms, sehingga memakan waktu dalam proses pencatatan transaksi.
Layanan (service)	Proses pemesanan tidak tepat guna karena pelanggan harus datang langsung ke Magetan.

Sistem yang telah bekerja sekarang ini dilakukan dengan cara manual atau kegiatan transaksi pemesanan dan penjualan yaitu melalui telepon atau datang langsung. Saat memesan produk maka karyawan akan meminta pembayaran secara DP (*down payment*) sebagai tanda jadi produksi dan membuat surat faktur pemesanan yang nantinya akan dikirim ke bagian produksi. Selanjutnya karyawan mencatat dan mengecek transaksi. Bagian produksi membuat produk yang diinginkan pelanggan. Setelah jadi, pelanggan bisa mengambil sendiri produk pesannya tersebut dengan datang langsung ke UKM.

Sistem yang telah berjalan memiliki beberapa kelemahan. Untuk mengukur kualitas penyampaian informasi dilakukan analisis dengan menggunakan metode PIECES (*performance, information, economic, control, efficiency, and service*). Pada penelitian [7] PIECES merupakan teknik pembelajaran terbaik dan inisiatif pengembangan sebuah sistem dengan memberikan pendekatan untuk memahami dan meningkatkan perawatan sebuah sistem. Di dalam PIECES terdapat enam komponen dalam menilai kepuasan pengguna terhadap sebuah sistem informasi yaitu: (1) *Performance*, (2) *Information*, (3) *Economics*, (4) *Control and Security*, (5) *Efficiency*, dan (6) *Service*. Adapun kelemahan yang telah dianalisis dengan menggunakan metode PIECES seperti pada Tabel 1.

3.2 ANALISIS KEBUTUHAN FUNGSIONAL

Spesifikasi aplikasi *e-commerce* yang akan diusulkan adalah sebagai berikut:

- Account**, setiap user atau pengguna dapat memiliki *account* dan melakukan *login*.
- Keranjang Belanja**, fungsi yang digunakan untuk menaruh koleksi yang telah dipilih oleh pengunjung, hal ini memungkinkan pengunjung dapat membeli produk lebih dari satu.
- Buku Tamu**, digunakan untuk meninggalkan pesan, saran, atau kritik mengenai *website*.

- Pencarian**, fungsi untuk melakukan pencarian koleksi.
- Info Produk**, informasi mengenai semua produk kerajinan kulit.
- Metode Pembayaran**, yaitu metode pembayaran yang dapat dipilih oleh pembeli, melalui kartu kredit atau tunai saat barang diantar untuk wilayah yang dapat dijangkau.
- Pemesanan**, fungsi yang disediakan bagi pelanggan yang ingin memesan produk kerajinan kulit sesuai dengan model yang diinginkan oleh pelanggan.

3.3 ANALISIS KEBUTUHAN NON-FUNGSIONAL

Analisis kebutuhan non-fungsional akan menjelaskan kebutuhan tentang *hardware, software, dan brainware*. Adapun rincian kebutuhan adalah sebagai berikut:

a. Hardware

- 1) Processor Core – Intel
- 2) Memory (RAM) – 8GB
- 3) Hardisk – Sandisk SSD 240GB
- 4) Network Speed – 1GBPs

b. Software

- 1) Apache – Version 2.4
- 2) Java
- 3) Unix System Operation
- 4) MySQL/MySQLi – Version 5.5.x
- 5) PHP – Version 5.4.x
- 6) Tomcat Version 8.5.3
- 7) SSL Certificated

c. Brainware

- 1) Administrator yang mampu mengoperasikan *e-commerce*
- 2) Owner yang mampu mengoperasikan aplikasi *e-commerce*

3.4 PERANCANGAN USE CASE DIAGRAM

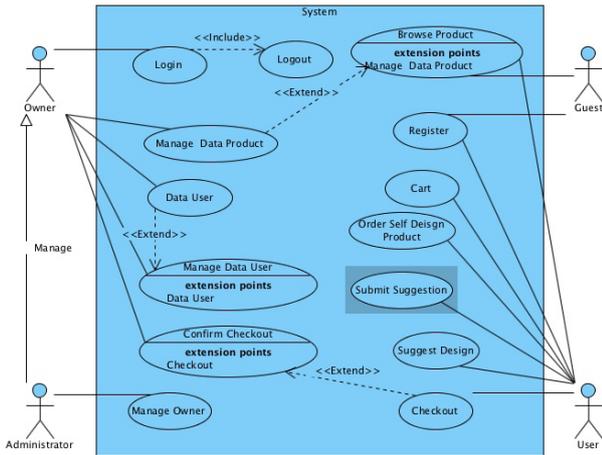
Pada tahap perancangan menggunakan model *object oriented*. UML (*Unified Modeling Language*) adalah 'bahasa' pemodelan untuk sistem atau perangkat lunak yang berparadigma 'berorientasi objek' [8]. Tujuan pemodelan (*modelling*) sistem adalah untuk melakukan penyederhanaan permasalahan-permasalahan yang kompleks sehingga dapat dengan mudah dipahami dan dipelajari.

Dalam UML terdapat diagram yang digunakan untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan aspek statistik dari sistem. Diagram struktur di UML terdiri atas:

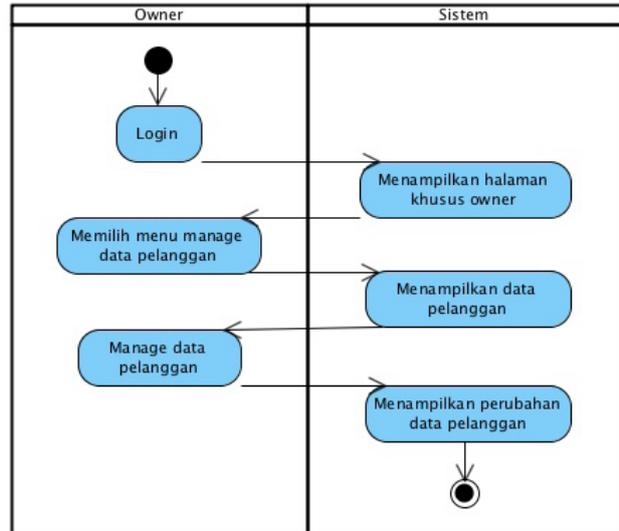
- Diagram Kelas (*Class Diagram*)
- Diagram Objek (*Object Diagram*)
- Diagram Komponen (*Component Diagram*)
- Diagram Deployment (*Deployment Diagram*)

Selain diagram struktur dalam UML, terdapat diagram untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan aspek dinamis dari sistem. Diagram perilaku di UML terdiri atas:

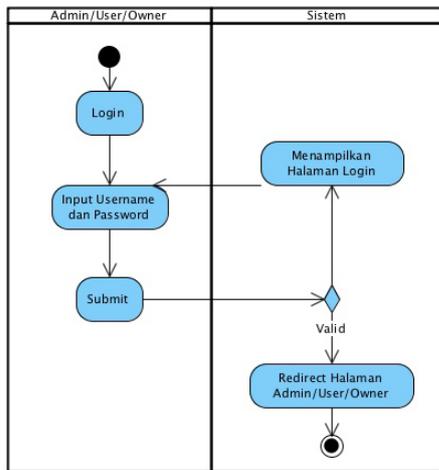
- Diagram Use Case (*Use Case Diagram*)
- Diagram Sekuen (*Sequence Diagram*)
- Diagram Kolaborasi (*Collaboration Diagram*)
- Diagram Statechart (*Statechart Diagram*)
- Diagram Aktivitas (*Activity Diagram*)



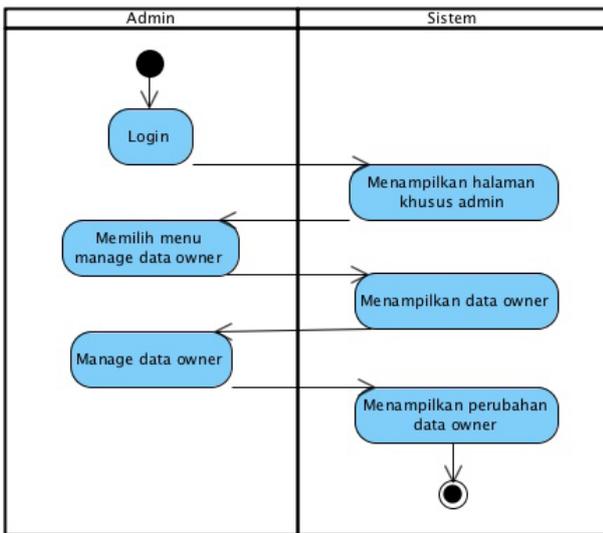
Gambar 6. Use Case Diagram



Gambar 9. Activity Diagram Manage Data Pelanggan



Gambar 7. Activity Diagram use case Login



Gambar 8. Activity Diagram Manage Data Owner

Pada penelitian ini akan dibangun sebuah rancangan dengan *use case diagram* yang akan menggambarkan perilaku setiap *actor* dalam mengimplementasikan aplikasi *e-commerce*. Menurut [9] *use case diagram* adalah sebuah skenario yang menggambarkan langkah-langkah yang menjelaskan interaksi antara pengguna dengan sistem untuk mencapai tujuan dari pengguna/*actor*. *Use case* adalah interaksi antara *actor* eksternal dan sistem, hasil yang dapat diamati oleh *actor*, berorientasi pada tujuan, dideskripsikan di diagram *use case* dan teks. Adapun desain *use case diagram* pada aplikasi *e-commerce* seperti pada Gambar 6.

Pada Gambar 6 terdapat 4 *actor* dengan masing-masing memiliki kegiatan yang saling berkaitan. Adapun *actor* yang terlibat adalah: (1) *Owner*, (2) *Administrator*, (3) *Guest*, dan (4) *User*. Kegiatan yang dilakukan oleh setiap *actor* sangat berpengaruh terhadap aktivitas *actor* lainnya. Sehingga integrasi setiap aktivitas dan integrasi data dapat tergambar dengan jelas pada Gambar 6.

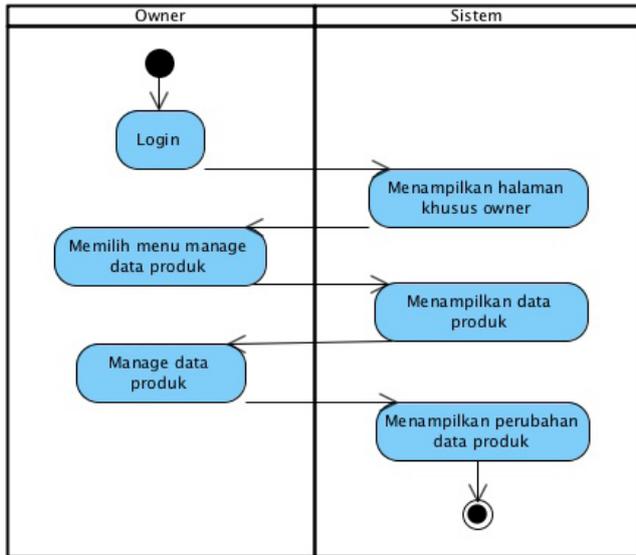
3.5 PERANCANGAN ACTIVITY DIAGRAM

Activity diagram merupakan jenis khusus dari *statechart* yang berfokus pada aktivitas-aktivitas, potongan-potongan dari proses yang berkorespondensi dengan metode-metode atau fungsi-fungsi anggota maupun aktivitas-aktivitas yang terjadi [9]. Pada penelitian ini desain alur setiap aktivitas yang terjadi digambarkan dengan *activity diagram*. Gambar di bawah ini akan menjelaskan rangkaian aktivitas yang terjadi.

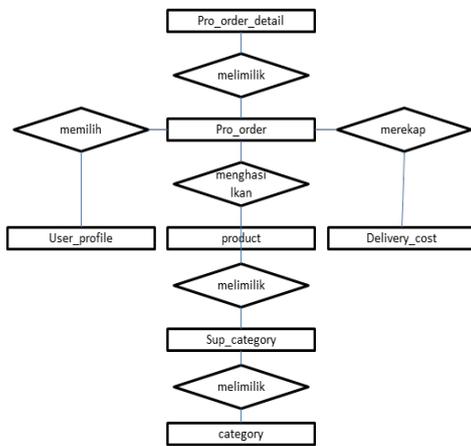
Gambar 7 menjelaskan *activity diagram* untuk *login* data pengguna. Hak akses *login* diberikan kepada administrator, *owner* (pemilik toko), dan pelanggan. Fungsi *login* diberikan untuk menjaga *security* data dan informasi yang disediakan.

Pada Gambar 8 menjelaskan *activity diagram* untuk proses kelola data *owner*. Pengelolaan data *owner* hanya dapat dilakukan oleh admin.

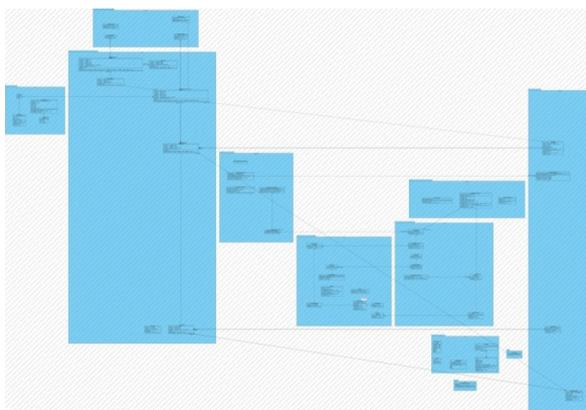
Pada Gambar 9 menjelaskan *activity diagram* untuk proses kelola data pelanggan. Pengelolaan data pelanggan dapat dilakukan oleh *owner*. Pelanggan yang telah melakukan registrasi pada aplikasi *e-commerce* dapat melakukan pemesanan dan pembelian produk.



Gambar 10. Activity Diagram Use Case Manage Data Produk



Gambar 11. ER-Diagram



Gambar 12. Class Diagram

Gambar 10 menjelaskan *activity diagram* untuk proses kelola data produk. Pengelolaan data produk dapat dilakukan oleh *owner*. *Owner* (pemilik toko) dapat memasukkan data produk sesuai dengan jenis dan kategori produk yang telah dihasilkan atau dimiliki.

3.6 PERANCANGAN MODEL DATA

Pada perancangan model data digambarkan dengan desain ERD yang menjelaskan relasi antarentitas yang

saling berkaitan pada aplikasi *e-commerce* pada konteks server. Sedangkan gambaran secara menyeluruh setelah memanfaatkan teknologi *paypal* digambarkan dengan desain *class diagram*.

a. Entity Relationship Diagram (ERD)

Pada Gambar 11 terdapat desain aliran relasi antarentitas yang menggambarkan proses *order* dan *delivery order*. Terdapat 7 entitas yang menggambarkan integrasi data antarentitas.

b. Class Diagram

Pengembangan sistem informasi tidak terlepas dari integrasi data. Data yang diolah menjadi sebuah informasi yang akan disajikan pada pengguna harus dirancang sedemikian rupa. Pada Gambar 12 menjelaskan desain integrasi data dengan menggunakan *class diagram*.

Class diagram ini menggambarkan desain sistem yang akan dibangun dalam *database*. Dapat dijadikan ilustrasi skema *database* yaitu dengan tabel-tabel yang akan digunakan. Setiap *class* akan diwakili oleh tabel, atribut, *class* yang akan menjadi *field* dalam tabel.

3.7 STRUKTUR TABEL

Gambar 12 terdapat *class-class* yang akan ditransformasikan menjadi beberapa tabel. Hasil transformasi tabel akan menghasilkan 9 tabel untuk aplikasi *e-commerce*. Adapun tabel-tabel tersebut adalah sebagai berikut:

a. Tabel App_User

Tabel 2. App_User

No	Field	Type	Length	Extra	Key
1	Id	Bigint	20	<i>Auto increment</i>	Primary Key
2	Sso_id	Varchar	30		Unique
3	Password	Varchar	100		
4	First_name	Varchar	30		
5	Last_name	Varchar	30		
6	Email	Varchar	30		
7	State	Varchar	30		

b. Tabel App_User_profile

Tabel 3. App_User_Profile

No	Field	Type	Length	Extra	Key
1	User_id	Bigint	20	<i>Auto increment</i>	Primary Key
2	User_profile_id	Bigint	20		Primary Key

c. Tabel Category

Tabel 4. Category

No	Field	Type	Length	Extra	Key
1	Cat_id	Int unsigned	10	<i>Auto increment</i>	Primary Key
2	Cat_name	Varchar	45		
3	Cat_desc	Varchar	200		

d. Tabel Delivery Cost

Tabel 5. Delivery_Cost

No	Field	Type	Length	Extra	Key
1	Delivery_id	Int unsigned	10	Auto increment	Primary Key
2	Zone_name	Varchar	45		
3	Delivery_cost	double			

e. Tabel Produk

Tabel 6. Produk

No	Field	Type	Length	Extra	Key
1	Pro_id	Int unsigned	10	Auto increment	Primary Key
2	Sub_cat_id	Int unsigned	10		Foreign Key
3	Pro_name	Varchar	45		
4	Pro_qty	Int unsigned	10		
5	Pro_price	double			
6	Pro_url	Varchar	200		
7	Pro_desc	Varchar	200		

f. Tabel Pro_Order

Tabel 7. Pro_Order

No	Field	Type	Length	Extra	Key
1	Order_id	Int unsigned	10	Auto increment	Primary Key
2	Pro_id	Int unsigned	10		Foreign Key
3	Id	Bigint	20		Foreign Key
4	Order_date	Date			
5	Order_qty	Int unsigned	10		
6	Total_price	Double			
7	Order_status	Varchar	45		
8	Delivery_id	Int unsigned	10		Foreign Key

g. Tabel Pro_Order_Detail

Tabel 8. Pro_Order_Detail

No	Field	Type	Length	Extra	Key
1	Order_details_id	Int unsigned	10	Auto increment	Primary Key
2	Order_id	Int unsigned	10		Foreign Key
3	Pro_id	Int unsigned	10		Foreign Key
4	Pro_qty	Int unsigned	10		

h. Tabel Sub_Category

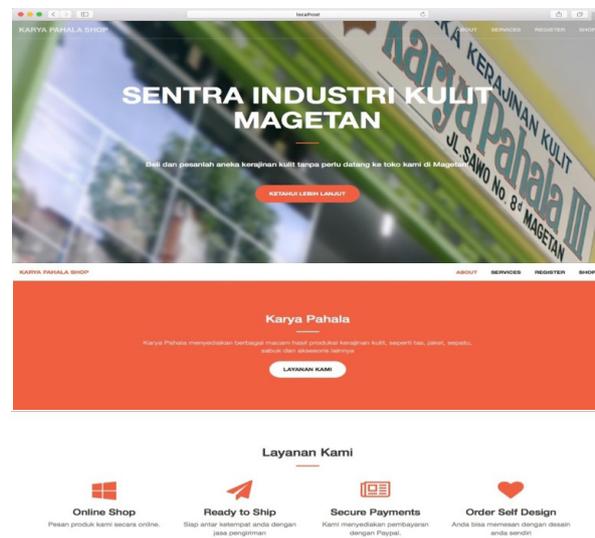
Tabel 9. Sub_Category

No	Field	Type	Length	Extra	Key
1	Subcat_id	Int unsigned	10	Auto increment	Primary Key
2	Cat_id	Int unsigned	10		Foreign Key
3	Sub_cat_name	Varchar	45		
4	Sub_cat_desc	Varchar	200		

i. Tabel User_profile

Tabel 10. User_Profile

No	Field	Type	Length	Extra	Key
1	Id	Bigint	20	Auto increment	Primary Key
2	Type	Varchar	20		Unique

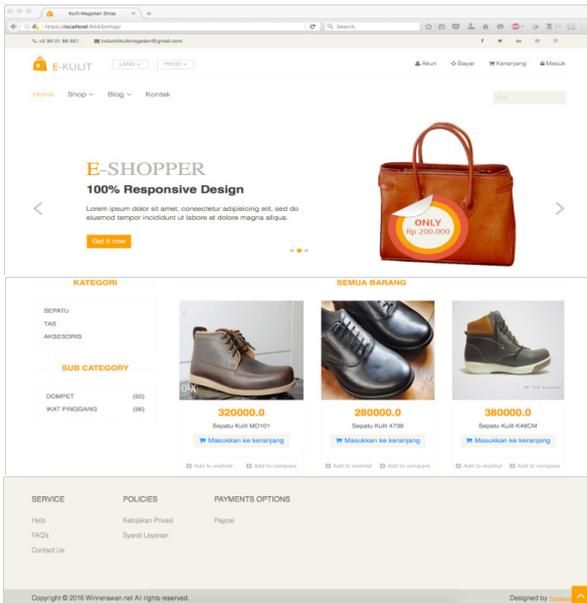


Gambar 13. Tampilan Antarmuka Home

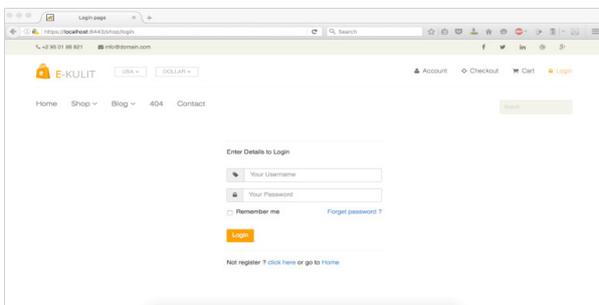
3.8 IMPLEMENTASI SISTEM

Pada tahap implementasi aplikasi *e-commerce* masih dijalankan secara *local*. Adapun tampilan *user interface* adalah sebagai berikut:

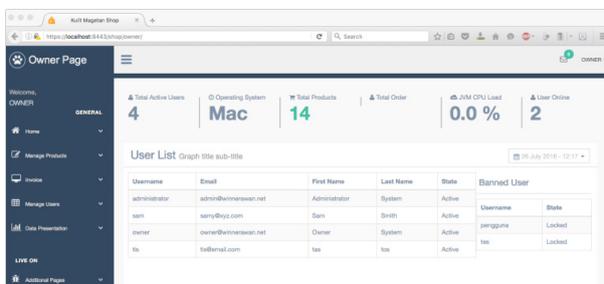
Gambar 13 menjelaskan tampilan antarmuka untuk halaman utama. Pada halaman ini memberikan informasi tentang UKM yang terdapat di sentra industri kulit Magetan dan contoh produk yang dimiliki masing-masing UKM.



Gambar 14. Halaman Kategori



Gambar 15. Halaman Login

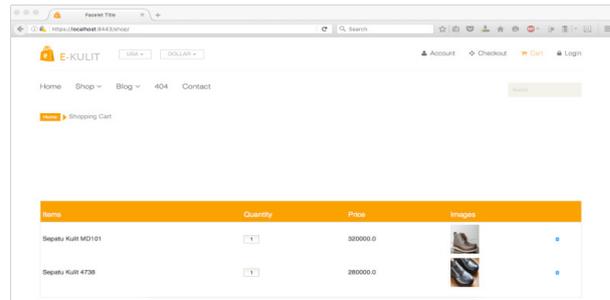


Gambar 16. Halaman Akses Owner

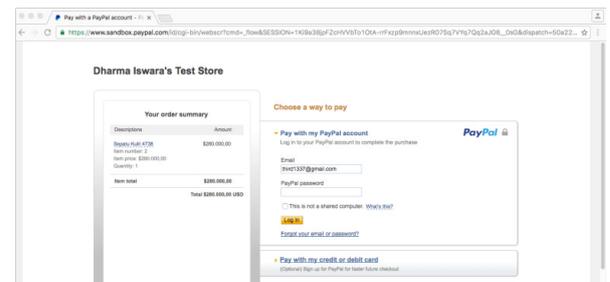
Gambar 14 memberikan informasi tentang kategori produk yang ada di UKM Karya Pahala (salah satu contoh UKM di Jalan Sawo). Informasi produk pada setiap katalog menjelaskan tentang kriteria produk beserta harga dan status ketersediaan produk.

Gambar 15 merupakan halaman antarmuka untuk melakukan login. Login dapat dilakukan oleh administrator, owner (pemilik toko), dan pelanggan.

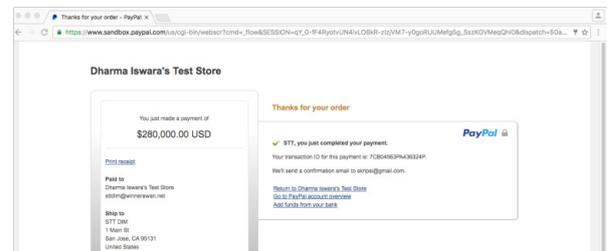
Gambar 16 merupakan halaman yang hanya dapat diakses oleh owner. Pada halaman ini owner dapat melakukan penambahan informasi dan penghapusan informasi terhadap produk yang dimiliki.



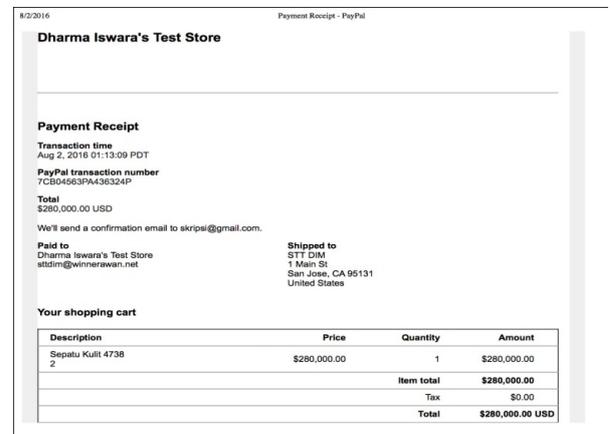
Gambar 17. Halaman Pembelian Online



Gambar 18. Halaman Login Paypal



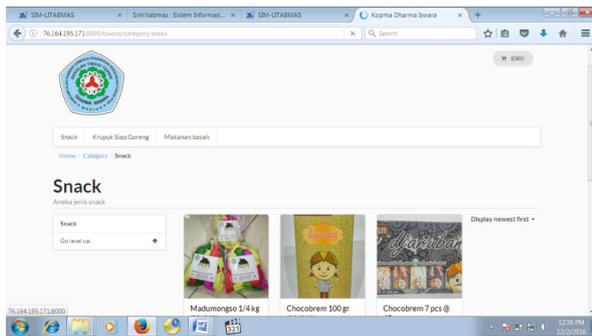
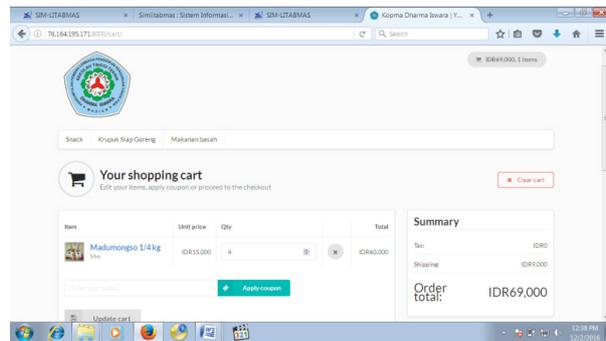
Gambar 19. Pembayaran dengan Paypal



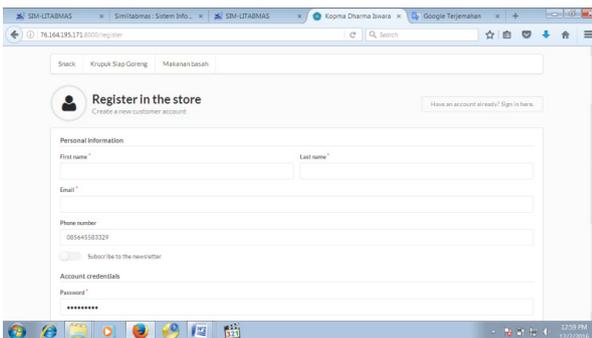
Gambar 20. Cetak Paypal

Gambar 17 memberikan informasi pembelian online yang dilakukan oleh pelanggan. Pelanggan dapat melakukan pemesanan produk yang diinginkan. Selain itu pelanggan juga dapat melakukan pembelian produk yang ada.

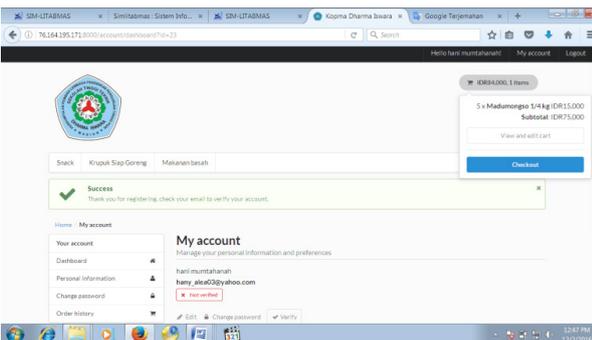
Gambar 18 merupakan halaman untuk pelanggan yang melakukan pembelian melalui aplikasi e-commerce. Pembelian secara online dapat dilakukan pembayaran dengan cara paypal. Pada halaman paypal ini pembeli bisa me-review ulang informasi tentang pembelian barang, paypal balance, serta alamat pengiriman yang dapat diubah.

Gambar 21. Halaman utama *e-commerce* Kopma Antik

Gambar 25. Halaman Konfirmasi Pembelian



Gambar 22. Halaman Registrasi Pelanggan



Gambar 23. Halaman Profil Pelanggan



Gambar 24. Halaman Pembelian Produk

Gambar 19 menampilkan informasi total pembayaran yang harus dikeluarkan oleh pelanggan. Selain itu terdapat beberapa informasi untuk melakukan pembayaran.

Setelah melakukan transaksi, maka sistem akan menampilkan bukti transaksi untuk pelanggan. Gambar 20 merupakan tampilan dari bukti transaksi untuk pelanggan.

3.9 Uji COBA SISTEM

Pada tahap selanjutnya adalah melakukan uji coba aplikasi *e-commerce*. Uji coba aplikasi bertujuan untuk melakukan penilaian terhadap tingkat kegunaan aplikasi *e-commerce* terhadap kebutuhan pengguna. Pengujian aplikasi dilakukan pada koperasi mahasiswa STT Dharma Iswara Madiun. Adapun instrumen antarmuka dalam uji coba adalah sebagai berikut:

4. KESIMPULAN

Pada penelitian ini menghasilkan sebuah rancang bangun sebuah aplikasi *e-commerce* untuk sentra industri kulit di Kabupaten Magetan. Aplikasi *e-commerce* ini akan menjadi salah satu penerapan teknologi sebagai media pemasaran hasil industri kulit yang ada di sentra industri kulit Kabupaten Magetan. Uji coba aplikasi pada Kopma Antik STT Dharma Iswara Madiun dilakukan untuk mengetahui tingkat kegunaan terhadap kebutuhan pengguna. Adanya aplikasi *e-commerce* ini mampu meningkatkan hasil pemasaran produk Kopma Antik STT sebanyak 15%. Di dalam tahap uji coba masih belum menggunakan metode yang akan menghasilkan sebuah nilai kuantitatif dari hasil uji coba. Pada penelitian selanjutnya akan dilakukan pengujian aplikasi dengan menggunakan teori *Nielsen* untuk mengetahui tingkat kegunaan aplikasi *e-commerce*.

5. DAFTAR PUSTAKA

- [1] Agustin, Eva M.S, "Persepsi Penggunaan Aplikasi Internet untuk Pemasaran Produk Usaha Kecil Menengah," in *Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, Yogyakarta, 2007, pp. J-13.
- [2] Haryanti, Sri. Tri Irianto, "Rancang Bangun Sistem Informasi E-Commerce Untuk Usaha Fashion Studi Kasus Omah Mode Kudus," *Jurnal Speed-Sentra Penelitian Engineering dan Edukasi*, vol. Vol. 3, no. 1, pp. 8-14, 2011.

- [3] Firdayanti Restika, "Persepsi Risiko Melakukan E-Commerce dengan Kepercayaan Konsumen dalam Membeli Produk Fashion Online," *Journal of Social and Industrial Psychology*, vol. I, no. 1, pp. 1-7, 2012.
- [4] Mumtahana, Hani Atun. Sekreningsih Nita, "Analisis Optimalisasi Penggunaan Teknologi Informasi dalam peningkatan Kualitas dan Pemasaran Produksi Kulit Magetan," in *Seminar Nasional Teknik dan Manajemen Industri*, Malang, 2015, pp. IV-22 - IV 27.
- [5] Jhon A. Zachman, *The Zachman Framework For Enterprise Architecture.*, 2003.
- [6] S Balaji and Dr. M. Sundararajan Murugaiyan, "Waterfall Vs V-Model Vs Agile : A Comparative Study On SDLC," *International Journal of Information Technology and Business Management*, vol. II, no. 1, pp. 26-29, June 2012.
- [7] Adi Supriyatna, "Analisis dan Evaluasi Kepuasan Pengguna Sistem Informasi Perpustakaan dengan Menggunakan PIECES Framework," *Jurnal Pilar Nusa Mandiri*, vol. XI, no. 1, pp. 43-52, Maret 2015.
- [8] Adi Nugroho, *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP*. Yogyakarta, Indonesia: Andi Yogyakarta, 2010.
- [9] Michael Yoseph Ricky, "Analisis dan Perancangan Sistem Informasi Laboratorium Rumah Sakit Kanker Dharmais dengan menggunakan Total Architecture Syntesis," *Comtech*, vol. I, no. 2, pp. 561-574, Desember 2010.

SISTEM PAKAR DENGAN METODE *DEMPSTER SHAFER* UNTUK DIAGNOSIS GANGGUAN LAYANAN INDIHOME DI PT TELKOM MAGELANG

Endang Lestari* , Emilya Uly Artha¹

¹Program Studi Teknik Informatika

Universitas Muhammadiyah Magelang

*endanglestari666@gmail.com

Abstrak

Indihome (Indonesia Digital Home) merupakan salah satu produk *triple play* dari PT Telkom berupa paket layanan telekomunikasi telepon rumah (*voice*), internet (*internet on fiber*), dan layanan tv kabel. Layanan Indihome berkembang cukup pesat. Namun demikian, kontinuitas akses layanan PT Telkom ini masih mengalami berbagai gangguan sehingga kualitas layanan belum optimal. Untuk mengatasi permasalahan tersebut maka dibuatlah aplikasi diagnosis gangguan layanan yang menggunakan metode *Dempster Shafer*, yaitu metode untuk mengukur nilai kepercayaan terhadap hasil analisis yang ditampilkan. Hasil penelitian ini adalah sebuah sistem pakar yang mendiagnosis gangguan layanan Indihome. Sistem secara otomatis memberikan hasil diagnosis dengan menampilkan jenis gangguan beserta solusinya berdasarkan gejala gangguan yang dialami. Kesimpulan penelitian ini adalah bahwa sistem pakar dengan menggunakan metode *Dempster Shafer* untuk menangani ketidakpastian data saat diagnosis gangguan layanan Indihome sangat membantu dalam mengatasi masalah kualitas pelayanan yang menurun.

Kata Kunci: sistem pakar, gangguan jaringan, diagnosis, *Dempster Shafer*.

1. Pendahuluan

Sistem pakar merupakan salah satu bidang teknik dari kecerdasan buatan yang dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja para pakar atau ahli [1]. Harapannya, orang biasa pun akan dapat menyelesaikan permasalahan yang dianggap cukup rumit yang tadinya hanya dapat diselesaikan oleh seorang pakar. Juga, terkadang seorang pakar tidak dapat melayani secara penuh karena terbatasnya waktu dan banyaknya hal yang harus dilayani sehingga sangat dibutuhkan sebuah sistem yang dapat menggantikan peran pakar tersebut [2].

Indihome (Indonesia *Digital Home*) merupakan salah satu layanan *triple play* dari produk Telkom berupa paket layanan telekomunikasi data yaitu telepon rumah (*voice*), internet (*internet on fiber*), dan layanan tv (*usestv cable*). Semua ini merupakan langkah berikutnya dalam pengembangan teknologi dari tembaga ke *fiber optic*. Penambahan jaringan *fiber optic* semakin diperluas demi melayani kebutuhan internet bagi pelanggan. Namun, tetap ada konsekuensi dari hal tersebut, yakni apabila layanan indihome mengalami gangguan maka fungsi komunikasi atau kepuasan pelanggan dapat berkurang, dan pelanggan belum tentu mengerti dan mengetahui cara memperbaikinya.

Telkom Magelang terdiri atas Daerah Telekomunikasi (Datel): Mungkid, Temanggung, Wonosobo, Purworejo, dan Kebumen. Setiap selesai dibangun jaringan baru di suatu wilayah, Telkom akan menambah tenaga baru yang

akan bertugas menangani perbaikan gangguan layanan Indihome. Untuk petugas eksisting yang biasa menangani gangguan tembaga (produk Speedy) diharapkan juga dapat menangani gangguan fiber (produk Indihome) untuk mengatasi gangguan migrasi dari tembaga ke fiber. Petugas baru biasanya akan bertanya tentang cara perbaikan kepada koordinator lapangan (korlap), dan ini akan menimbulkan kesulitan apabila si petugas baru itu belum memahami dasar-dasarnya. Untuk itu akan cukup bermanfaat bila dibangun suatu sistem pakar untuk membantu Sumber Daya Manusia (SDM) baru, petugas migrasi jaringan dari tembaga ke fiber, dan petugas yang masih kurang berpengalaman dalam menangani gangguan layanan Indihome. Diharapkan sistem ini akan memberi penyelesaian sesuai Standard Operation Procedure (SOP) yang ada.

Penelitian ini mencoba menggunakan metode *Dempster Shafer*, suatu teori matematika tentang pembuktian berdasarkan fungsi kepercayaan (*belief function*) dan pemikiran yang masuk akal (*plausible reasoning*). Metode ini digunakan untuk menggabungkan potongan-potongan informasi yang terpisah atau bukti-bukti untuk mengkalkulasi kemungkinan dari suatu peristiwa [3, 4].

2. Tinjauan Pustaka

Terdapat beberapa penelitian yang memiliki kemiripan dengan topik penelitian yang dilakukan penulis. Penelitian yang dilakukan Ferdian *et al.* melaporkan bahwa mereka

telah mendafta queri-queri yang telah terbentuk (yakni berupa pertanyaan-pertanyaan yang merupakan gejala-gejala kerusakan sambungan telepon) dan membuat sistem informasi yang akan memberikan prediksi kerusakan sambungan telepon [5].

Penelitian yang dilakukan Suhendra *et al.* mengungkapkan bahwa penggunaan metode Forward Chaining dalam pembuatan Sistem Pakar Diagnosa Gangguan Sambungan Telepon bisa dikatakan mampu memberikan pelayanan yang lebih efisien bagi pengguna. Ini dicapai dengan cara menggunakan rancangan-rancangan sistem dan basis pengetahuan yang dikelola untuk menjadi sebuah sistem pakar, didukung penyampaian yang memakai bahasa yang baik dan benar sehingga memudahkan pengguna [6].

Penelitian yang dilakukan Panigrahi *et al.* menyebutkan penerapan teori *Dempster Shafer* yang digabungkan dengan metode pengambilan keputusan Bayesian (*Bayesian inferencing*). Kedua metode digabungkan untuk melakukan deteksi terjadinya penyalahgunaan telekomunikasi seluler. Deteksi didasarkan pada data perilaku pelanggan telepon seluler saat ini dan riwayatnya di masa lalu. Metode *Dempster Shafer* digunakan untuk mendeteksi apakah panggilan telepon bersifat normal, abnormal, atau mencurigakan. Jika dideteksi sebagai mencurigakan, tingkat kepercayaan diperbesar atau diperkecil dengan metode Bayesian dengan menghitung similaritas panggilan dengan data riwayat yang telah ada. Penggabungan kedua metode mampu mengurangi alarm palsu (*false alarm*) namun tetap mampu mendeteksi panggilan abnormal [7].

Ketiga penelitian yang relevan di atas memiliki kesamaan dengan penelitian penulis. Kesamaan dengan penelitian pertama dan kedua adalah pada objek penelitiannya, yakni tentang layanan telekomunikasi. Sedangkan kesamaan dengan penelitian ketiga yaitu algoritma yang digunakan, metode *Dempster Shafer*.

3. Analisis dan Perancangan Sistem

3.1 Analisis

Analisis sistem dimengerti sebagai penguraian sistem informasi yang utuh ke dalam bagian-bagian dengan maksud mengidentifikasi masalah dan mengevaluasi kebutuhan-kebutuhan. Dari analisis ini diharapkan akan muncul usulan perbaikan.

Basis Pengetahuan (*Knowledge Base*) mengandung pengetahuan terkait pemahaman, formulasi, dan penyelesaian masalah. Kemampuan sebuah sistem pakar dalam memecahkan suatu masalah diperoleh dengan cara merepresentasikan pengetahuan seorang atau beberapa orang pakar ke dalam format tertentu dan menyimpannya dalam basis pengetahuan. Sebuah sistem pakar berbasis aturan umumnya menggunakan aturan (*rules*) untuk merepresentasikan pengetahuan di dalam basis pengetahuannya.

Tabel 1 menunjukkan basis pengetahuan yang digunakan dalam penelitian ini. Tabel memperlihatkan keluhan-keluhan terkait layanan Indihome milik Telkom.

Tabel 1. Basis Pengetahuan

Jenis Gangguan	Gejala	Solusi
TLP mati total	TLP mati tidak ada nada	Solusi yang diberikan apabila telepon mati tidak ada nada adalah cek dengan test <i>phone</i> kemungkinan pesawat telepon pelanggan yang rusak, jika jaringan telepon, rekonek kabel UTP telepon, cek pada <i>splitter</i> pastikan RJ 11 yang masuk pada <i>phone</i> dan modem tidak terbalik. Apabila tlp di paralel, cek pada sambungan <i>rosette</i> pastikan tidak lembab air. Jika masih belum bisa coba ganti <i>splitter</i> dan <i>rosette</i> baru, atau ganti kabel dan <i>connector</i> RJ 11 yang baru.
	Cek rosette lembab air	
	Cek splitter tidak berfungsi	
	Cek kabel UTP TLP rusak	
Kabel UTP TLP rusak	Cek connector RJ 11 tidak berfungsi	Pertama kali cabut RJ 11 pada pesawat telepon, kemudian pasang kembali. Jika masih belum bisa, dilakukan penggantian kabel UTP TLP yang baru.
	Nada TLP ngetut-tut-tut	
IP PC tidak sesuai dengan IP modem	Nada TLP nada panjang	Cek koneksi <i>local area</i> , lakukan ping tes dari pc langsung ke modem, cek perangkat pelanggan sudah mendapat IP modem, jika belum bisa dilakukan isi IP secara manual. <i>Restart</i> modem atau <i>reset</i> modem, kemudian <i>setting</i> ulang konfigurasi.
	Internet tidak bisa <i>connect</i> (TBC)	
Terisolir	Cek PC tidak mendapatkan IP	Buka aplikasi SAMS, lakukan bukis (buka isolir) dengan memasukkan nomor internet pelanggan.
	Internet tidak bisa <i>connect</i> (TBC)	
Spam, virus	Cek tidak ada tunggakan pembayaran	Meng-offkan semua <i>firewall</i> , pastikan antivirus tidak <i>blocking</i> koneksi.
	Koneksi lambat PC pelanggan di <i>share</i>	

Jenis Gangguan	Gejala	Solusi
Profil paket kuota habis	Koneksi lambat <i>Bandwidth</i> kecil Cek paket internet kuota	Cek pada aplikasi spins untuk cek sisa kuota, edukasi pelanggan.
Kabel <i>patchcord</i> rusak	Koneksi putus-putus Cek modem normal Cek kabel <i>patchcord</i> rusak	Ganti kabel <i>patchcord</i> baru, melakukan penyambungan ulang dengan kabel <i>dropcore</i> .
DNS / Proxy	Tidak bisa <i>browsing</i> (TBB) Cek IP PC sesuai dengan IP modem	Bisa dilakukan cek/ganti <i>proxy</i> yang ada di <i>browser</i> . Mengubah DNS, masukkan DNS google (<i>Preferred DNS Server 8.8.8.8, Alternate DNS Server 8.8.4.4</i>).
Profil di <i>port</i> tidak sesuai paket	Tidak bisa <i>browsing</i> (TBB) Cek <i>bandwidth</i> tidak sesuai paket Cek konfigurasi seting hilang	<i>Setting-an create</i> pada modem hilang, lakukan <i>create</i> ulang, masuk pada <i>putty</i> , masukkan <i>script create setting</i> konfigurasi dan masukkan data nomor telepon, nomor internet, dan SN (<i>Serial Number</i>) modem.
Kabel <i>dropcore</i> rusak	Lampu indikator PON mati Cek konfigurasi <i>setting</i> hilang	Cek ukuran/redaman kabel dengan menggunakan OPM <i>meter</i> (ukuran baik tidak >25dBm), lalu lakukan rekoneksi sambungan/pergantian <i>dropcore</i> . <i>Setting create</i> pada modem hilang, lakukan <i>create</i> ulang, masuk pada <i>putty</i> , masukkan <i>script create setting</i> konfigurasi dan masukkan data nomor telepon, nomor internet, dan SN (<i>Serial Number</i>) modem.
<i>Fast connector</i> rusak	Lampu indikator LOS merah Cek <i>fast connector</i> tidak berfungsi/rusak	Dilakukan penggantian <i>fast connector</i> baru, kemudian lakukan penyambungan ulang
Adaptor rusak / modem rusak	Lampu indikator POWER mati Modem ONT tidak menyala	Dilakukan pengecekan dengan adaptor dan modem <i>test</i> , apabila tidak berfungsi dapat diganti dengan adaptor atau modem ONT baru.
<i>Maintenance server</i>	Tidak bisa membuka web tertentu Cek proxy sudah sesuai Cek dns sudah sesuai	Kemungkinan terdapat perbaikan server pada alamat web yang dituju.
Kabel RCA / HDMI rusak	Gambar UseeTv Blank Cek konfigurasi sudah sesuai	Dilakukan penggantian kabel RCA/HDMI baru.
Alamat dari server otentikasi tidak sesuai	<i>Error 1302</i> Koneksi ke EPG gagal	Jika gangguan <i>error 1302</i> . Disebabkan alamat <i>homepage EPG (Electronic Program Guide)</i> gagal terhubung atau alamat dari server otentikasi tidak sesuai. Dicoba dengan <i>reboot</i> STB (<i>set top box</i>) dengan cara: pencet tombol SET pada <i>remote useetv</i> > pilih konfigurasi> isi <i>password</i> dengan 6321> tingkat lanjut> <i>reboot</i> . Pastikan kabel UTP dari STB menancap di LAN Port 4 modem, agar mendapatkan alamat dari server yaitu ip 10 .x.x.x

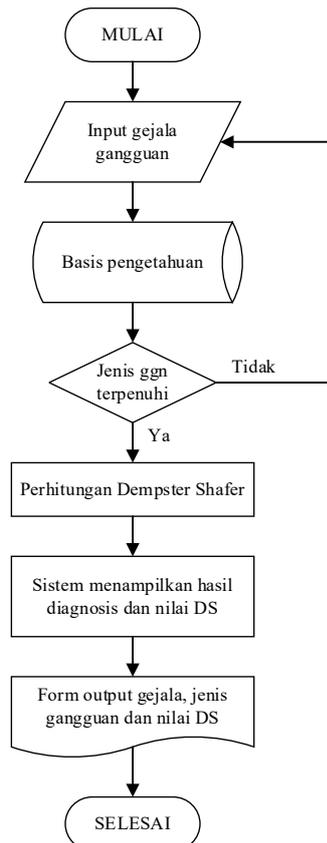
Jenis Gangguan	Gejala	Solusi
Profil DHCP tidak valid	<i>Error</i> 1305 Cek <i>username</i> tidak sesuai Cek <i>password</i> tidak sesuai	Jika gangguan <i>error</i> 1305. Kemungkinan parameter DHCP (<i>Dinamic Host Configuration Protocol</i>) dari modem tidak benar. Dilakukan <i>reboot</i> modem seperti gangguan 1302, dilanjutkan dengan cek info jaringan: pilih menu tingkat lanjut> pilih sistem informasi> pilih info jaringan> pastikan ip mendapatkan ip 10.x.x.x dengan memastikan parameter otentikasi, nomor internet, dan <i>password usestv</i> sudah benar.
Kabel UTP LAN / Connector RJ 45 rusak	<i>Error</i> 1901 Koneksi ke EPG gagal	Jika gangguan <i>error</i> 1901. Disebabkan kabel jaringan tidak tersambung. Periksa koneksi fisik dari kabel jaringan. Cek koneksi kabel UTP & RJ 45 dari modem ke arah STB, coba di- <i>reconnect</i> ulang. Apabila kabel UTP & RJ 45 sudah tidak berfungsi dapat dilakukan penggantian baru.
Konfigurasi vlan <i>multicast</i> hilang	<i>Error</i> 4514 <i>Channel multicast</i> mengalami data <i>timeout</i>	Jika gangguan <i>error code</i> 4514 sama dengan gangguan <i>channel live</i> TV tidak muncul. Cek pada <i>embassy</i> , pastikan data Rx Power untuk OLT dan ONU tidak > 25 dBm. Cek konfigurasi <i>vlan multicast</i> apabila konfigurasi hilang atau <i>channel multicast</i> mengalami data <i>timeout</i> , maka <i>create</i> lagi dengan memasukkan <i>script</i> konfigurasi <i>vlan</i> pada <i>putty</i> , dengan mendaftarkan nomor internet dan SN (<i>serial number</i>) modem.
Konfigurasi <i>setting</i> hilang	Gambar <i>UseTv</i> putus-putus Cek konfigurasi <i>setting</i> hilang	Jika kualitas gambar tv putus-putus, tekan tombol <i>info</i> pada <i>remote control</i> > panah kanan [<i>volume +</i>] > tampil <i>signal power</i> & <i>quality standard</i> >70%. Atau dilakukan <i>create</i> ulang konfigurasi.
Kebinding	Penggantian modem STB baru <i>Error</i> 70116204	Jika gangguan <i>error code</i> 70116204, disebabkan <i>account</i> pelanggan di <i>lock</i> , cek pada aplikasi <i>embassy</i> jika kebinding maka di <i>unbinding</i> , terutama untuk modem baru.
Salah <i>username</i> / <i>password</i>	<i>Error</i> 70116206 Cek <i>username</i> tidak sesuai Cek <i>password</i> tidak sesuai	Jika gangguan <i>error code</i> 70116206, disebabkan <i>username</i> /nomor rekening dan <i>password</i> salah pada settingan menu STB, cek pada aplikasi SOAP <i>username</i> dan <i>password</i> usestv yang benar> <i>setting</i> ulang kembali di STB.
Modem STB rusak	Modem STB tidak menyala atau tidak berfungsi Lampu indikator LINK mati	Jika lampu indikator LINK mati, Cek kelayakan kabel UTP, <i>connector</i> RJ 45 dari ONT ke STB. Cek dengan modem <i>test</i> , apabila STB rusak dapat dilakukan penggantian modem baru.

3.2 Perancangan Sistem

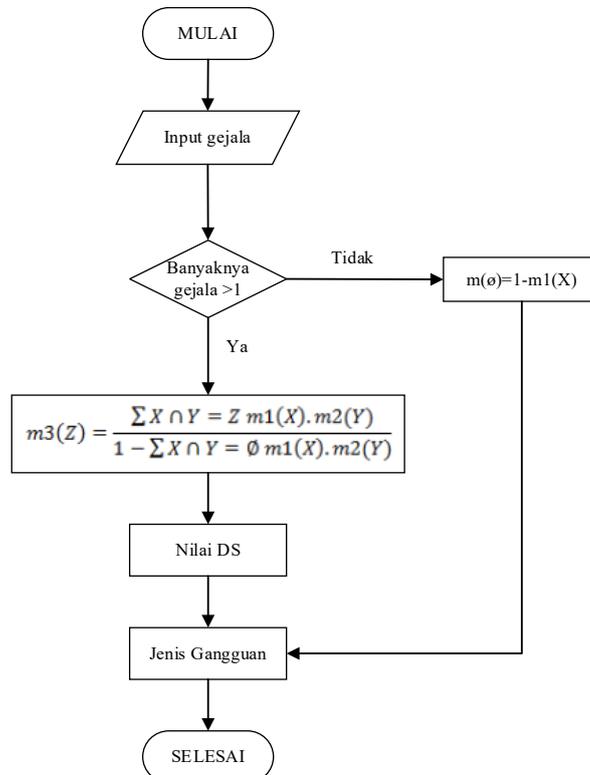
a. Diagram alur konsultasi

Diagram alur konsultasi merupakan alur yang digunakan oleh petugas gangguan untuk melakukan

konsultasi dengan pelanggan, berdasarkan gejala-gejala gangguan. Diagram alur konsultasi disajikan pada Gambar 1 yang dimulai dari input gejala gangguan hingga ditampilkannya *output* gejala dan jenis gangguan.

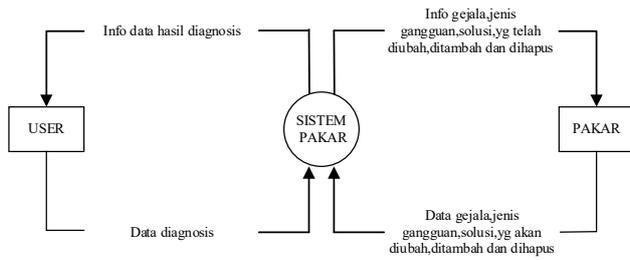


Gambar 1. Diagram Alur Konsultasi

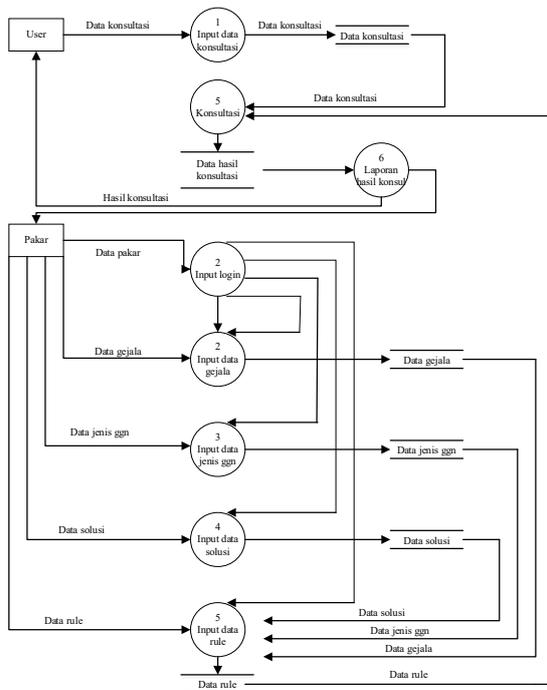
Gambar 2. Diagram Alur Algoritma *Dempster Shafer*

b. Diagram Alur Algoritma *Dempster Shafer*

Diagram alur algoritma *Dempster Shafer* merupakan alur dari perhitungan untuk menentukan nilai kepercayaan (lihat Gambar 2).



Gambar 3. Diagram Konteks



Gambar 4. DFD

c. Perancangan Basis Data

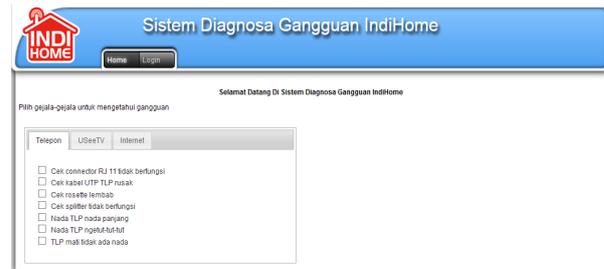
Basis data dirancang mengikuti diagram konteks. Diagram konteks merupakan gambaran umum mengenai proses yang terjadi dalam sistem. Diagram konteks terdiri dari entitas dan proses. Entitas merupakan unsur luar dari sistem yang mendapat dan memberi data ke sistem, sedangkan proses merupakan kegiatan pengolahan data yang dijalankan di dalam sistem. Diagram konteks untuk sistem yang dibuat pada penelitian ini diperlihatkan pada Gambar 3. Selanjutnya diagram konteks dijabarkan lebih detail dalam bentuk *Data Flow Diagram* (DFD) yang disajikan pada Gambar 4.

3.3 Implementasi dan Pengujian

a. Implementasi Sistem

Implementasi sistem merupakan tahap menerjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh mesin komputer, serta penerapan perangkat lunak pada keadaan siap untuk dioperasikan.

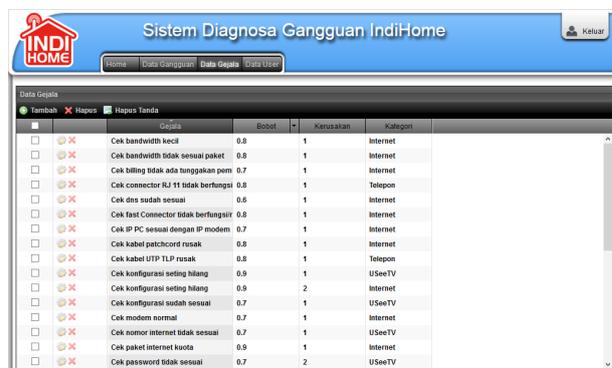
Perangkat lunak yang digunakan dalam implementasi adalah Windows 7 sebagai sistem operasi, Notepad++ sebagai editor, PHP MySQL, dan Google Chrome. Sedangkan kebutuhan perangkat keras adalah komputer dengan P4 Processor 2 GHz, RAM 512 GHz, Hardisk 40 Gb, Monitor 15”, dilengkapi *Keyboard* dan *Mouse*.



Gambar 5. Menu Utama



Gambar 6. Menu Login



Gambar 7. Menu Tambah Gejala



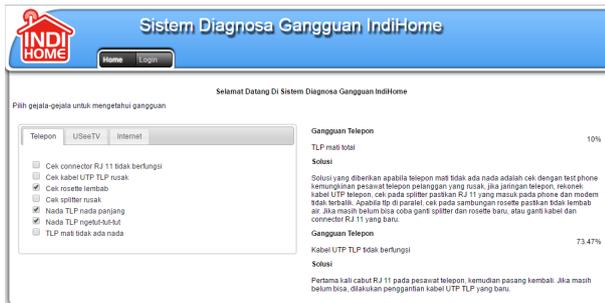
Gambar 8. Menu Tambah Jenis Gangguan

b. Implementasi Interface

Antarmuka dibuat menggunakan bahasa PHP dan hasilnya ditampilkan di Gambar 5-8.

c. Pengujian

Pengujian sistem dilakukan dengan cara mencoba program aplikasi dengan memasukkan data ke dalam form-form yang ada pada sistem. Tahap ini merupakan kelanjutan tahap implementasi. Pengujian sistem termasuk pengujian program secara menyeluruh. Kumpulan program yang telah terintegrasikan perlu menjalani uji coba untuk melihat apakah sebuah program dapat menerima, memproses, dan memberikan keluaran program dengan baik sesuai rancangan.



Gambar 9. Checklist Gejala

Gambar 9 merupakan pengujian berbeda rule yaitu:

- Jika Nada TLP ngetut-tut-tut (rule sama)
- Dan Nada TLP nada panjang (rule sama)
- Dan Cek rosette lembab (rule berbeda)
- Maka TLP mati total, ds = 10%
- Maka Kabel UTP TLP tidak berfungsi, ds = 73%

Pengujian ini menghasilkan keluaran dua jenis gangguan: TLP mati total dan Kabel UTP TLP tidak berfungsi. Apabila gejala dalam satu rule maka penghitungan akan digabungkan, apabila berbeda rule maka akan dihitung sendiri dengan kemungkinan bobot kepercayaan yang berbeda

d. Penghitungan Dempster Shafer

Perhitungan *Dempster Shafer* didemonstrasikan dalam contoh perhitungan berikut.

- Jika Nada TLP ngetut-tut-tut (rule sama), bel=0,9
- Dan Nada TLP nada panjang (rule sama), bel=0,8

Maka :

$$P01 (\text{bel}) = 0,9$$

$$P01 (\Phi) = 1 - 0,9 = 0,1$$

$$P02 (\text{bel}) = 0,8$$

$$P02 (\Phi) = 1 - 0,8 = 0,2$$

$$DS = 0,9 * 0,8$$

$$1 - (0,1 * 0,2)$$

$$= 0,72 / 1 - 0,02$$

$$= 0,72 / 0,98$$

$$= 0,73 \text{ atau } 73\%$$

Dan Cek rosette lembab (rule berbeda), bel=0,9

Maka :

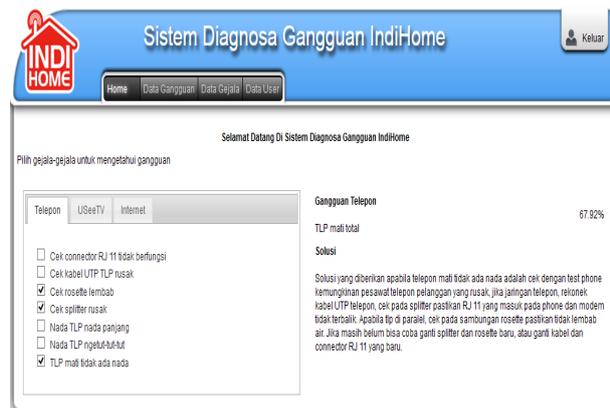
$$P01 (\text{bel}) = 0,9$$

$$P01 (\Phi) = 1 - 0,9 = 0,1 \text{ atau } 10\%$$

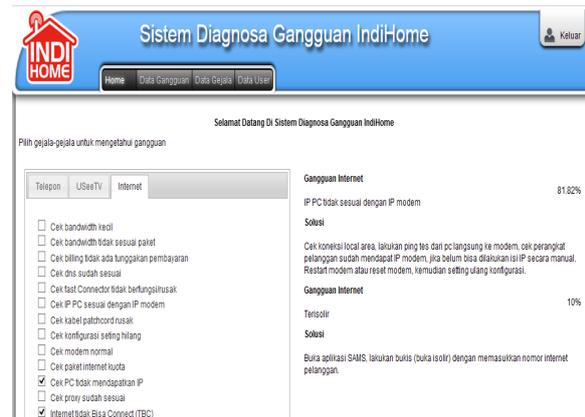
4. Hasil dan Pembahasan

4.1 Hasil

Hasil dari Sistem Pakar Diagnosis Gangguan Layanan Indihome dibahas di bawah ini, untuk dua kejadian gangguan.



Gambar 10. Konsultasi Gangguan Telepon



Gambar 11. Konsultasi Gangguan Internet

Gambar 10 memperlihatkan contoh user menginputkan gejala gangguan yaitu telepon mati tidak ada nada, cek rosette lembab, dan cek splitter rusak. Setelah dilakukan proses dan perhitungan, diperoleh keluaran hasil konsultasi yaitu dengan penghitungan *Dempster Shafer* dengan bobot kepercayaan untuk jenis gangguan telepon mati total sebesar 67% disertai dengan solusi penyelesaian.

Gambar 11 menunjukkan tampilan aplikasi ketika user menginputkan gejala gangguan berupa cek PC tidak mendapatkan IP dan internet tidak bisa connect (tbc). Hasil perhitungan oleh aplikasi memberikan keluaran dua hasil konsultasi dengan penghitungan *Dempster Shafer*. Keputusan yang dipilih adalah keluaran dengan bobot kepercayaan yang tertinggi, yaitu jenis gangguan IP, yakni PC tidak sesuai dengan IP modem. Bobot kepercayaan untuk keputusan itu adalah adalah 81%.

4.2 Pembahasan

Bagian ini membahas sistem informasi yang telah dibuat.

a. Hasil diagnosis

Proses diagnosis dua contoh yang terkait Gambar 10 dan Gambar 11 terlihat bahwa sistem menampilkan hasil akhir konsultasi. Pada tahapan hasil ini ditampilkan jenis gangguan, nilai *Dempster Shafer*, dan solusinya.

b. Kesesuaian dengan rule

Pada contoh kasus yang disampaikan di atas, diperoleh hasil yang sesuai dengan *rule* dan proses inferensi sistem yang digunakan. Adapun *rule* yang menuji jenis gangguan telepon mati total (yakni IP

PC tidak sesuai dengan IP modem, alamat dari server otentikasi tidak sesuai) adalah sebagai berikut:

- a. JIKA Cek *rosette* lembab
DAN Cek *splitter* rusak
DAN Telepon mati tidak ada nada
MAKA Telepon mati total
 - b. JIKA Cek PC tidak mendapatkan IP
DAN Internet tidak Bisa *Connect* (TBC)
MAKA IP PC tidak sesuai dengan IP modem
 - c. JIKA *Error 1302*
DAN Koneksi ke EPG gagal
MAKA Alamat dari server otentikasi tidak sesuai
- c. Perhitungan *Dempster Shafer*
- a) JIKA Cek *rosette* lembab, bel 0,9
DAN Cek *splitter* rusak, bel 0,8
DAN Telepon tidak ada nada, bel 0,9
MAKA Telepon mati total

Perhitungan :

Gejala 1 = P01 (bel) = 0,9

$$P01 (\Phi) = 1-0,9 = 0,1$$

Gejala 2 = P02 (bel) = 0,8

$$P02 (\Phi) = 1-0,8 = 0,2$$

$$DS = \frac{0,9*0,8}{1-(0,1*0,2)}$$

$$= 0,72 / 1-0,02$$

$$= 0,72 / 0,98$$

$$= 0,73$$

Gejala 3 = P03 (bel) = 0,9

$$P03 (\Phi) = 1-0,9 = 0,1$$

$$H01 = 0,73$$

$$H01 (\Phi) = 1-0,7 = 0,3$$

$$DS = \frac{0,73*0,9}{1-(0,3*0,1)}$$

$$= 0,65 / 1-0,03$$

$$= 0,65 / 0,97$$

$$= 0,67$$

Jadi bobot kepercayaan untuk jenis gangguan telepon mati total yaitu 67%

- b) JIKA Cek PC tidak mendapatkan IP, bel 0,9
DAN Internet tidak Bisa *Connect* (TBC), bel 0,9
MAKA IP PC tidak sesuai dengan IP modem

Perhitungan :

Gejala 1 = P01 (bel) = 0,9

$$P01 (\Phi) = 1-0,9 = 0,1$$

Gejala 2 = P02 (bel) = 0,9

$$P02 (\Phi) = 1-0,9 = 0,1$$

$$DS = \frac{0,9*0,9}{1-(0,1*0,1)}$$

$$= 0,81 / 1-0,01$$

$$= 0,81 / 0,99$$

$$= 0,81$$

Jadi bobot kepercayaan untuk jenis gangguan "IP PC tidak sesuai dengan IP modem" bernilai 81%

- c) JIKA *Error 1302*, bel 0,9
DAN Koneksi ke EPG gagal, bel 0,9
MAKA Alamat dari server otentikasi tidak sesuai

Perhitungan :

Gejala 1 = P01 (bel) = 0,9

$$P01 (\Phi) = 1-0,9 = 0,1$$

Gejala 2 = P02 (bel) = 0,9

$$P02 (\Phi) = 1-0,9 = 0,1$$

$$DS = \frac{0,9*0,9}{1-(0,1*0,1)}$$

$$= 0,81 / 1-0,01$$

$$= 0,81 / 0,99$$

$$= 0,81$$

Jadi bobot kepercayaan untuk jenis gangguan "alamat dari server otentikasi tidak sesuai" bernilai 81%

Pembahasan beberapa paragraf terakhir menunjukkan bahwa hasil diagnosis yang ditampilkan oleh aplikasi sudah sesuai dengan *rule* yang dipilih. Selanjutnya kesesuaian *rule* dapat dibuktikan dengan perhitungan *Dempster Shafer*. Dengan demikian aplikasi ini telah berjalan sesuai dengan rancangan yang direncanakan.

Sistem ini dapat membantu petugas magang dan petugas eksisting di jaringan tembaga dalam menangani gangguan Indihome apabila koordinator lapangan sibuk atau sulit dihubungi. Hal ini karena keputusan pakar dapat ditiru dalam tingkat tertentu oleh sistem.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Sistem telah berhasil dibangun dan dapat membantu petugas magang dan petugas eksisting di jaringan tembaga dalam menangani gangguan indihome apabila koordinator lapangan tidak dapat dihubungi

Metode *Dempster-Shafer* berhasil diimplementasikan dalam sistem pakar diagnosis gangguan layanan Indihome. Sistem dapat digunakan untuk mendiagnosis jenis-jenis gangguan dengan memasukkan gejala-gejala yang dialami pelanggan lewat antarmuka sistem.

Perhitungan *Dempster-Shafer* untuk dua contoh dengan dua gejala terbukti sesuai dengan keputusan pakar. Untuk jenis gangguan kabel UTP Telepon rusak (gejala pertama dengan nilai *belief* 0,8 dan kedua dengan nilai *belief* 0,9) menghasilkan bobot kepercayaan 73%. Sehingga dengan menginputkan 2 gejala dapat diketahui jenis gangguan dan bobot kepercayaannya berdasarkan nilai kepercayaan.

5.2 Saran

Sistem pakar ini menggunakan metode ketidakpastian *Dempster Shafer*, yakni metode berdasarkan fungsi kepercayaan (*belief*) dan pemikiran yang masuk akal (*plausible*), dengan input gejala minimal 2 gejala untuk kemudian dapat dijalankan penghitungan DS yang sesuai. Metode lain dapat diujicoba untuk menyelesaikan ketidakpastian, misal menggunakan *certainty factor*, dan *best first search*.

6. Daftar Pustaka

- [1] Shu-Hsien Liao, "Expert System Methodologies and Applications-a decade review from 1995 to 2004," *Expert systems with applications*, vol. 28, no. 1, pp. 93-103, 2005.

- [2] Kusrini, *Sistem Pakar Teori dan Aplikasi*. C.V Andi Offset, Yogyakarta, 2006.
- [3] F.T.M. Aditya, *Sistem Pakar Untuk Mendiagnosis Kerusakan Handphone dengan Metode Dempster Shafer*. Skripsi. Universitas Muhammadiyah Magelang, 2014.
- [4] Hamid Reza Pourghasemi, Biswajeet Pradhan, Candan Gokceoglu, and Kimia Deylami Moezzi, "A Comparative Assessment of Prediction Capabilities of Dempster-Shafer and Weights-of-Evidence Models in Landslide Susceptibility Mapping Using GIS," *Geomatics, Natural Hazards and Risks*, vol. 4, no. 2, pp. 93-118, 2013.
- [5] F. Erhan, F. Jaka, Parmahaki, P. Raden, *Sistem Pakar Mengidentifikasi Gangguan Sambungan Telepon PT.TELKOM*. Makalah Penelitian Ilmu Komputer. Universitas Pakuan, Bogor, 2004.
- [6] Suhendra, I.B. Ramdan, A. Faisal, T.L. Iga, Y. Dewi, "Sistem Pakar Diagnosa Gangguan Sambungan Telepon," *Seminar Nasional Teknologi Informasi dan Multimedia di STMIK AMIKOM*, Yogyakarta, 2015.
- [7] Suvasini Panigrahi, Amlan Kundu, Shamik Sural, and Arun K. Majumdar, "Use of Dempster-Shafer theory and Bayesian Inferencing for Fraud Detection in Mobile Communication Network," in *Australian Conference on Information Security and Privacy*, Australia, 2007, pp. 446-460.

Identifikasi Kadar Amoniak sebagai Indikator Bau Toilet Menggunakan Perangkat MAS TUQUL

Salsabila Nurulfarah Mahmudah^{1*}, Heni Hanifah¹, Wahyu Titis Satria Utama²,
Ismi Kamelia Najib Putri¹, Rahmat Kurniawan³, Husni Thamrin¹

¹Program Studi Informatika

Univeritas Muhammadiyah Surakarta

²Program Studi Teknik Elektro

Universitas Muhammadiyah Surakarta

³Program Studi Teknik Kimia

Universitas Muhammadiyah Surakarta

*l200160132@student.ums.ac.id

Abstrak

Toilet merupakan tempat untuk melakukan ekskresi. Kegiatan ekskresi ini terdapat 2 wujud zat, yaitu berupa zat cair atau urin dan keringat, dan zat padat berupa feses. Urin manusia menyuplai setidaknya 1% terhadap kuantitas air limbah domestik dengan 80% nitrogen dan 50% fosfat. Kebutuhan untuk melakukan ekskresi menjadikan pentingnya toilet di setiap tempat, baik di rumah maupun di tempat umum. Kebutuhan itu perlu diimbangi dengan tersedianya toilet layak pakai, yang memperhatikan kebersihan, kenyamanan dan kesehatan bagi pengguna. Kenyamanan toilet, bau toilet umum, dapat diidentifikasi dengan mengetahui kadar amoniak. Hubungan antara kadar amoniak dengan bau di toilet dapat dinyatakan dengan persamaan $Y = 1.8045 + 0.5836 X$. Y merupakan kategori bau, sedangkan X merupakan kadar amoniak (ppm). Kadar amoniak menyumbang 67% dari keseluruhan bau yang ada di toilet. Kadar amoniak yang telah disebutkan dalam Peraturan Kesehatan, yakni sebesar 2 ppm. Nilai tersebut termasuk kategori bau biasa. Pada pengukuran berdasarkan bau amoniak ini, Mas Tuqul dapat membacanya dengan benar. Mas Tuqul dapat membaca secara stabil selama 15 menit. Dari hasil pembacaan tersebut, dapat dilihat dengan mudah kadar bau yang ada di dalam toilet. Diharapkan dengan adanya Mas Tuqul ini dapat mempermudah untuk mengetahui keadaan udara yang ada di toilet.

Kata Kunci: toilet, amoniak, monitoring, Mas Tuqul.

1. Pendahuluan

Toilet merupakan tempat untuk melakukan aktivitas ekskresi (mengeluarkan zat sisa metabolisme). Setiap hari, rata-rata orang melakukan aktivitas buang air besar satu kali dan buang air kecil sebanyak 6–7 kali [1] dengan rata-rata volume urin yang dikeluarkan sebanyak 400-2000 mililiter [2] [3]. Ekskresi yang dilakukan oleh manusia mempunyai dua wujud zat, yaitu berupa zat cair atau urin dan keringat, serta zat padat berupa feses. Urin manusia menyuplai setidaknya 1% terhadap kuantitas air limbah domestik dengan 80% nitrogen dan 50% fosfat [4]. Kebutuhan untuk melakukan ekskresi menjadikan pentingnya toilet di setiap tempat, baik di rumah maupun di tempat umum. Kebutuhan itu perlu diimbangi dengan tersedianya toilet layak pakai, yang memperhatikan kebersihan, kenyamanan, dan kesehatan bagi pengguna.

Menurut Peraturan Menteri Pekerjaan Umum Nomor 24 tahun 2008 tentang Pedoman Pemeliharaan dan Perawatan Bangunan Gedung (lihat tabel pada

Gambar 1), suatu toilet disebut layak jika toilet bersih dan ruangnya tidak berbau amis, pesing, ataupun anyir [5]. Peraturan Menteri Kesehatan Republik Indonesia Nomor 8 Tahun 2014 mengatur tentang pelayanan kesehatan SPA (Sehat Pakai Air). Peraturan itu menyatakan bahwa toilet dikategorikan nyaman jika toilet tersebut mempunyai suhu berkisar antara 22-25 derajat celsius (°C) dan kelembaban berkisar antara 40-70% [6].

Namun, banyak toilet di Indonesia yang jauh dari kata layak. Banyak masyarakat baik WNI (Warga Negara Indonesia) ataupun WNA (Warga Negara Asing) yang ada di Indonesia, mengeluhkan kuantitas dan kualitas toilet umum yang ada [7]. Menurut *World Economic Forum*, kualitas toilet di Indonesia berada pada peringkat 34 dari 144 negara yang diamati [8]. Dengan demikian, kualitas toilet di Indonesia masih relatif rendah. Senada dengan itu, Badan Pusat Statistik menunjukkan bahwa akses sanitasi yang layak di Indonesia masih sebesar 55.6% [9] dan salah satu unsur sanitasi itu adalah toilet.

STANDAR KEBERSIHAN	
DIFUSER/GRILL	: Bersih, tidak bernoda, tidak ada sarang laba-laba, tidak berdebu
PLAFOND	: Bebas dari kotor, tidak ada noda, tidak berdebu, tidak ada sarang laba-laba
KACA	: Bersih, jelas, bening, tidak ada noda, tidak ada kotoran, tidak berdebu, frame kaca bersih.
HORIZONTAL BLIND	: Bersih, tidak kotor, tidak berdebu, rapi.
SAKLAR & STOP KONTAK	: Tidak berdebu, tidak bernoda
FURNITURE	: Bersih, tidak berdebu, tidak bernoda, bila diusap tidak membekas, tidak ada sampah, tidak ada sarang laba-laba.
LANTAI	: Bersih, tidak berdebu, tidak bernoda, tidak buram, tidak basah, tidak bau, nat lantai bersih.
KARPET	: Bersih, tidak berdebu, tidak bernoda, tidak bau, tidak basah, tersisir rapi.
TOILET	
- RUANG	: Tidak bau : amis, pesing, anyir.
- KACA CERMIN	: Bening, terang, tidak kusam, tidak bernoda, tidak basah.
- KLOSED	: Mengalir lancar, tidak ada noda, tidak ada bercak air disekelilingnya, tidak bau.
- KRAN	: Tidak berkarat, tidak basah, tidak kusam.
- LANTAI	: Bersih, kering, tidak ada noda, tidak ada sampah.
- H. PINTU	: Bersih, tidak ada noda, mengkilat (sesuai aslinya).
- URINOIR	: Bersih, tidak ada noda, tidak bau, tidak berkarat.
- KESET	: Tidak berdebu, tidak basah, tidak ada sampah, tidak bau.
TANGGA	
- RAILING	: Tidak berdebu, tidak ada noda, bila diusap tidak membekas.
- BORDES	: Tidak berdebu, tidak ada sampah, tidak basah, tidak bau.
- P. BESI	: Tidak berdebu, tidak bernoda, tidak ada bercak.
TAMAN	: Subur, bersih, rapi, indah.
LANTAI ASPAL	: Bersih tidak ada sampah, tidak banjir, tidak kotor tanah.

Gambar 1. Cuplikan tabel tentang Standar Kebersihan versi Departemen Pekerjaan Umum [5]

Rendahnya kualitas toilet menimbulkan ketidaknyamanan dan menjadi sumber bakteri yang berbahaya bagi kesehatan. "Toilet tidak higienis merupakan tempat bersarangnya berbagai jenis kuman yang berkembang biak dan berpotensi menjadi sumber penyakit membahayakan," kata dr Rouli Nababan SpA [10]. Setiap 20 menit kuman berkembang biak dan dalam jangka 24 jam 1 sel dapat menjadi 8 juta sel. Berbagai penyakit dapat ditimbulkan, seperti diare, hepatitis A, cacangan, ISPA, tifus, disentri, bahkan infeksi saluran kencing. Belum lagi dengan masalah sanitasi yang mungkin menyebarkan *scabies*, *lice*, dan cacang penyebab kaki gajah.

Tulisan ini membahas pengembangan perangkat yang diberi nama MAS TUQUL (*Monitoring Ammonia System To Upgrade Quality*) yang dapat memantau kandungan amoniak dalam toilet, dengan hipotesis amoniak merupakan material yang signifikan untuk mengindikasikan kualitas toilet. Pemantauan parameter dimaksudkan untuk menjadi bagian dari sistem pemeliharaan kebersihan toilet sehingga diharapkan dapat membantu meningkatkan kualitas toilet khususnya toilet publik. Selanjutnya tulisan ini menguraikan metode pengembangan perangkat, hasil pengembangan perangkat, pengujian di lapangan, dan pembahasan hasil pengujian. Pada pembahasan hasil pengujian, kami menganalisis data-data tersebut, menggunakan metode regresi linier sederhana dan signifikan atau tidaknya amoniak ditunjukkan dengan nilai koefisien determinasi.

2. Metode

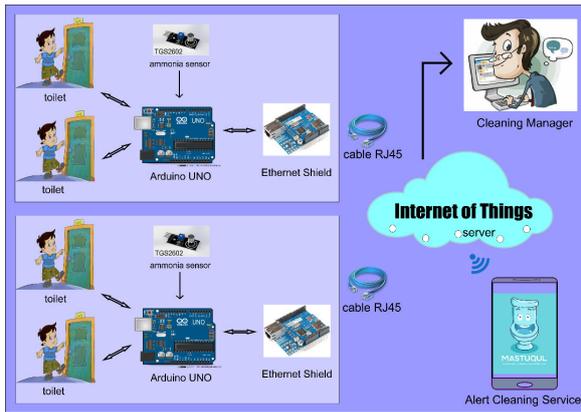
Perangkat keras yang digunakan dalam proses pengembangan sistem adalah Sensor TGS2602, *Power Supply Mini* MB102, Adaptor 12V, Kabel RJ45, *Acrylic*, *Ethernet shield*, dan Arduino UNO. Perangkat lunak yang digunakan untuk pembangunan alat terdiri atas komputer dengan sistem operasi Windows/Linux, *kompiler* bahasa C dan Arduino IDE.



Gambar 2. Diagram alir pelaksanaan

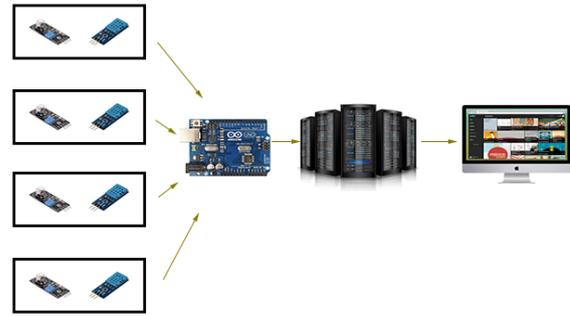
Pengembangan alat melalui enam tahapan sebagaimana ditunjukkan Gambar 2. Masing-masing tahapan dijelaskan sebagai berikut:

- Studi Literatur Kegiatan ini dilakukan dengan mencari penunjang dari jurnal ilmiah atau makalah. Bahan-bahan yang dicari meliputi dua langkah berikut:
 - Metode pembacaan kadar amoniak pada sensor Metode penghitungan kadar amoniak pada sensor merupakan hal dasar penting yang digunakan pada alat ini. Menemukan rumus yang sesuai dengan melakukan penafsiran grafik yang terdapat pada *data sheet* sensor TGS2602.
 - Batas ambang untuk kadar amoniak bagi manusia Batas ambang suatu zat kimia mempunyai kadar-kadar yang berbeda. Untuk amoniak dapat ditemukan di dalam peraturan menteri kesehatan no. 8 tahun 2014 [6].

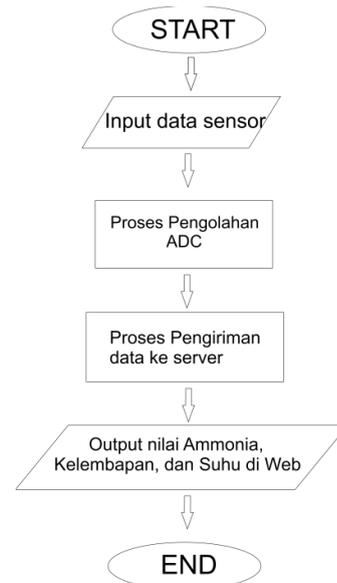


Gambar 3. Skema penerapan/peletakan alat

- c) **Identifikasi dan Pemodelan Perangkat**
 Alat ini disusun menggunakan 2 jenis benda utama, yakni 1 jenis sensor dan sebuah mikrokontroler. Sensor tersebut adalah sensor TGS2602. Terdapat banyak sensor yang bisa digunakan untuk mengukur kadar amoniak, namun sensor yang memiliki rentang yang sesuai dengan kebutuhan penelitian kami adalah sensor TGS2602, yakni rentang di antara 0-20 ppm. Untuk sistem yang digunakan, kami menggunakan data yang berasal dari sensor yang berupa sinyal digital. Data tersebut dikirim melalui *Ethernet shield*, diolah terlebih dulu di servernya dan hasilnya akan ditampilkan pada web yang sudah ada. Skema konfigurasi alat diperlihatkan pada Gambar 3.
- d) **TGS2602**
 TGS2602 merupakan sebuah sensor yang bisa digunakan untuk mendeteksi beberapa macam gas kimia. Di antaranya adalah hidrogen sulfida, amoniak, etanol, toluen, air, dan hidrogen. Rentang kepekaan terhadap bau secara keseluruhan cukup besar, yakni mulai dari 0.1 hingga 100 ppm.
- e) **Arduino UNO**
 Arduino UNO merupakan sebuah mikrokontroler berbasis ATmega328. Mempunyai 14 pin digital *input/output*, 6 *input* analog, resonator keramik 16 MHz, koneksi USB, *jack* listrik, *header* ICSP, dan tombol *reset*. Arduino di sini berguna untuk mengubah sinyal analog menjadi sinyal digital dan mengirim data ke server yang telah diterima oleh sensor.
- f) **Server**
 Server merupakan sebuah tempat yang berfungsi untuk menyimpan data yang telah dikirimkan oleh sensor.
- g) **Toilet**
 Toilet adalah sebuah tempat yang digunakan untuk mengeluarkan zat-zat sisa hasil pencernaan. Dari zat-zat yang telah dikeluarkan, didapat data berupa angka-angka yang menunjukkan tingkat kebauannya berdasar kadar amoniak.



Gambar 4. Alur pengiriman data



Gambar 5. Diagram alir konsep program

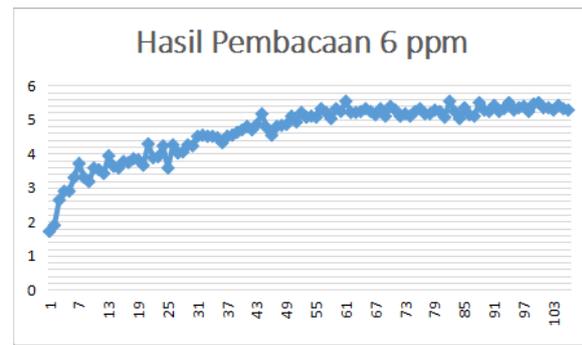
- b. **Perancangan Perangkat**
 Selain terdapat 2 benda yang utama, terdapat juga beberapa benda pelengkap, yakni resistor, *motherboard*, kabel jumper, kabel RJ45, dan *Ethernet Shield*. Perangkat ini akan menjadi 2 kelompok alat (lihat Gambar 4). Kelompok pertama berfungsi untuk mengambil data. Kelompok satunya untuk mengolah dan mengirimkan data.
- c. **Pembuatan Perangkat**
 Pembuatan alat ini memudahkan koordinator ataupun manager petugas kebersihan untuk mengontrol kelayakan yang ada. Penggunaannya juga cukup mudah, hanya menekan tombol pada alat tersebut, keadaan toilet akan terdokumentasi dan data akan langsung disimpan dalam server. Alur pembuatan perangkat tampak pada Gambar 5.
 Cara kerja alat ini adalah sebagai berikut:
- Sensor TGS2602 terhubung dengan arduino
 - Sensor TGS2602 mendeteksi amoniak
 - Nilai amoniak yang terdeteksi diubah ke digital melalui *Analog Digital Converter* yang sudah ada di arduino
 - Nilai amoniak yang terdeteksi dikirim ke server menggunakan *Ethernet Shield*
 - Server menampilkan Nilai Amonia yang telah diterima

- d. Pengujian Perangkat
- Pengujian Sensor
Pengujian sensor dimaksudkan untuk melakukan proses kalibrasi atau konfirmasi terhadap fungsi sensor. Pengujian dilakukan dengan membuat satu set cairan dengan kadar yang telah ditentukan. Pembacaan pada sensor dicermati dan dipastikan setara seperti kadar cairan yang telah ditentukan.
 - Pengujian *Software*
Sensor yang sudah dirangkai, diuji untuk menentukan interval pengambilan data, durasi penilaian kualitas toilet, dan bagaimana cara menilainya. Nilai yang sudah didapat dari sensor akan dibandingkan dengan batas ambang kadar amoniak. Apabila nilai dari sensor lebih tinggi dibanding dengan batas ambang yang ada, maka petugas *cleaning service* akan mendapat notifikasi untuk membersihkan kamar mandi. Pemberitahuan berupa pesan singkat melalui SMS *Gateway* yang dikirimkan langsung dari server.
 - Pengujian Semua Perangkat
Prototipe dirangkai terlebih dahulu. Alat yang dirangkai terdiri dari 1 sensor, yakni sensor TGS2602, arduino, dan *ethernet shield*. Rangkaian sensor diletakkan pada suatu ruangan yang sudah didesain seperti keadaan udara yang ada di toilet. Sedangkan arduino yang berlaku sebagai pengolah data diletakkan jauh dari ruangan. Antara arduino dan sensor dihubungkan dengan kabel RJ45. Selain itu, pengujian dilakukan untuk mengetahui kekurangan pada alat dan sistem yang telah dibuat. Tahap ini dilakukan untuk mengetahui kekurangan apa saja yang terdapat pada alat dan sistem yang dibuat.
- e. Analisis
Analisis dilakukan terhadap hasil pembacaan data kadar amoniak di beberapa toilet kampus UMS dengan menggunakan regresi linier sederhana. Kadar amoniak merupakan variabel X sedangkan bau toilet yang diukur dengan menggunakan skala *likert* menjadi variabel Y.

3. Hasil dan Pembahasan

3.1 Kalibrasi perangkat MAS TUQUL

Proses pengujian ketepatan pembacaan sensor dilakukan dengan melakukan kalibrasi. Menguji coba sensor tersebut dengan amonia yang telah ditentukan ukurannya. Pengukuran dilakukan melalui beberapa kali percobaan. Dan hasil dari pengukuran tersebut adalah sensor sudah bisa untuk membaca kadar amoniak sesuai dengan ukurannya. Hasil pembacaan sensor dapat dilihat pada grafik Gambar 6. Sumber vertikal menunjukkan nilai ppm dari cairan sedangkan sumbu horizontal menunjukkan waktu pembacaan. Setelah lebih dari satu menit, hasil pembacaan sensor sudah mendekati sama dengan nilai konsentrasi bahan yang diukur, yaitu sebesar 6 ppm.



Gambar 6. Hasil pembacaan sensor

3.2 Amoniak sebagai indikator

Untuk mengetahui seberapa besar hubungan kadar amoniak terhadap bau toilet, dilakukan analisis regresi linier sederhana. Variabel Y menunjukkan tingkat bau pada skala *likert* dengan makna sebagai berikut: 1. Wangi sekali, 2. Wangi, 3. Biasa, 4. Bau, 5. Bau sekali. Skala *likert* merupakan variabel ordinal yang ditunjukkan dengan Y. Variabel X merupakan kadar amoniak pada satuan ppm. Koefisien regresi linier diestimasi dengan metode *least square*. Metode itu mengharuskan analisis dilakukan menggunakan variabel interval. Oleh karena itu, variabel Y harus diubah menjadi skala interval, dengan menggunakan metode *successive interval*.

Telah dilakukan 16 pengamatan yang menghasilkan data seperti pada Tabel 1. Pada pengamatan, kadar amoniak ditambah secara perlahan (variabel X pada tabel). Kemudian persepsi responden diamati melalui pernyataan tingkat bau. Setelah diperoleh data, selanjutnya dilakukan analisis regresi linier yang hasilnya ditunjukkan pada Gambar 7.

Tabel 1. Nilai variabel X pada skala Likert

Nomor pengamatan	Variabel Y	Y Transformasi	Variabel X
1	2	1	0.05
2	2	1	0.06
3	2	1	0.06
4	2	1	0.05
5	3	2.271106291	0.1
6	3	2.271106291	0.1
7	3	2.271106291	0.08
8	3	2.271106291	0.07
9	3	2.271106291	0.08
10	3	2.271106291	0.09
11	3	2.271106291	0.08
12	3	2.271106291	0.07
13	4	3.542212581	2.7
14	4	3.542212581	2.9
15	4	3.542212581	3
16	4	3.542212581	3.3

SUMMARY OUTPUT								
<i>Regression Statistics</i>								
Multiple R	0.81880309							
R Square	0.67043849							
Adjusted R Square	0.64689838							
Standard Error	0.55160867							
Observations	16							
<i>ANOVA</i>								
	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>			
Regression	1	8.66588	8.66588	28.48069	0.000105			
Residual	14	4.25981	0.304272					
Total	15	12.92569						
	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>	<i>Lower 95.0%</i>	<i>Upper 95.0%</i>
Intercept	1.80451297	0.163282	11.05149	2.67E-08	1.454307	2.154719	1.454307	2.154719
X Variable 1	0.58369766	0.109374	5.33673	0.000105	0.349115	0.818281	0.349115	0.818281

Gambar 7. Hasil analisis regresi linier

Tabel 2. Prediksi tingkat bau berdasarkan kadar amoniak

Y	1	2	3	4	5
X (ppm)	0	0.33	2.04	3.76	5.47

Hasil analisis regresi menunjukkan hubungan antara tingkat bau menurut persepsi manusia dan kadar amoniak, yaitu $Y = 1.8045 + 0.5836 X$ dengan koefisien determinasi 67%. Kadar amoniak mempengaruhi 67% dari bau toilet. Sedangkan 33% bau disumbangkan dari bahan selain amoniak.

3.3 Pembahasan

Setelah diperoleh persamaan regresi, dapat dilakukan prediksi kadar amoniak untuk setiap kategori bau toilet sebagaimana disajikan di Tabel 2. Entri tabel 2 diperoleh dengan memasukkan nilai variabel Y dan menghitung nilai variabel X yang setara.

Dengan adanya korelasi antara kadar amoniak dan persepsi bau oleh manusia, dapat diprediksi tingkat kenyamanan toilet berdasarkan kadar amoniak yang terukur. Jika diinginkan bahwa toilet tidak boleh dipersepsikan sebagai “bau” dan “bau sekali”, maka ambang batas kadar amoniak yang terukur tidak boleh lebih dari 3,76 ppm, sesuai nilai pada Tabel 2. Jika kadar amoniak melebihi batas tersebut, maka toilet harus segera dibersihkan.

Perangkat Mas Tuqul masih dalam proses pengembangan, yaitu untuk memanfaatkannya dalam manajemen kebersihan toilet di suatu kawasan atau gedung. Kemampuan perangkat Mas Tuqul memprediksi bau toilet dengan mengikuti persepsi manusia menjadi penting. Jika sensor ini diletakkan di berbagai toilet pada sebuah gedung, maka pengawas kebersihan gedung dapat melakukan pemantauan terhadap kebersihan toilet. Jika ada toilet yang melebihi ambang batas, maka perangkat Mas Tuqul akan memberikan notifikasi kepada petugas yang akan segera membersihkan toilet. Pengawas kebersihan dan bahkan manajer gedung juga dapat melihat rekaman data dalam sehari atau seminggu atau dalam rentang waktu tertentu dan mencari tahu apakah toilet pernah terbengkalai dalam arti bau melebihi ambang batas dalam waktu lama.

Standar kementerian kesehatan menunjukkan bahwa ambang batas toilet yang sehat adalah kadar amoniak tidak boleh melebihi 2 ppm [6]. Angka kadar amoniak ini termasuk kategori “normal” menurut hasil penelitian penulis. Setidaknya menurut persepsi responden penelitian ini, kadar amoniak sedikit melebihi batas menurut kementerian kesehatan, masih dikategorikan normal oleh responden.

4. Kesimpulan

Berdasarkan penelitian yang dilakukan, dapat disimpulkan bahwa perangkat Mas Tuqul dapat digunakan untuk mendeteksi bau toilet. Bahan yang dijadikan tolok ukur adalah kadar amoniak yang terkandung dalam udara di toilet. Hasil pengukuran kadar amoniak dapat dikonversi menjadi persepsi tingkat bau. Perangkat Mas Tuqul ini bisa digunakan untuk mengelola kebersihan toilet di suatu kawasan atau gedung.

5. Ucapan Terima Kasih

Terima kasih disampaikan kepada Direktorat Jenderal Pembelajaran dan Kemahasiswa (Belmawa), Kemenristekdikti yang telah memberikan pendanaan sehingga kegiatan penelitian ini bisa berjalan.

6. Referensi

- [1] Chaunie Brusie. (2017, Pebruari) Health Line. [Online]. <http://www.healthline.com/health/how-often-should-you-pee>
- [2] Eggert Moeller, J. F. McIntosh, and D. D. Van Slyke, “Studies of Urea Excretion. II: Relationship Between Urine Volume and the Rate of Urea Excretion by Normal Adults,” *Journal of Clinical Investigation*, vol. 6, no. 3, p. 427, 1928.
- [3] Leann Mikes. (2015, April) Live Strong. [Online]. <http://www.livestrong.com/article/516316-what-is-normal-urinary-output-for-an-adult/>
- [4] Yuyun L Rofiqoh and Eddy S Soedjono, “Studi Potensi Urin Manusia Hasil Composting Toilet dalam Sistem Ecological Sanitation (ECOSAN) (Studi Kasus Puskota Surabaya),” Institut Teknologi Sepuluh Nopember, Surabaya, Skripsi 2010.

- [5] Departemen Pekerjaan Umum. (2011) Peraturan Menteri Pekerjaan Umum tentang Pedoman Pemeliharaan dan Perawatan Bangunan Gedung. [Online]. <http://www.pu.go.id/uploads/services/2011-12-01-12-59-56.pdf>
- [6] Kementerian Kesehatan Republik Indonesia. (2014) Peraturan Menteri Kesehatan Republik Indonesia Nomor 8 Tahun 2014 tentang Pelayanan Kesehatan SPA. [Online]. <http://www.peraturan.go.id/permen/kemenkes-nomor-8-tahun-2014.html>
- [7] Hendi Setiawan. (2015, Juni) Kompasiana. [Online]. http://www.kompasiana.com/hendisetiawan/kenapa-toilet-umum-kebanyakan-kotor_551ad19ca333115820b659a7
- [8] Klaus Schwab, "The Global Competitiveness Report 2014-2015," *World Economic Forum*, 2014.
- [9] Badan Pusat Statistik. (2016) Badan Pusat Statistik. [Online]. <https://www.bps.go.id/linkTabelStatistik/view/id/1537>
- [10] Sri Noviarni. (2014, Oktober) Koran Sindo. [Online]. <http://lifestyle.sindonews.com/read/930083/152/toilet-kotor-sumber-penyakit-1417143870/10>
- [11] Anto Dajan, "Pengantar Metode Statistik Jilid II", *Lembaga Penelitian, Pendidikan, dan Penerangan Ekonomi dan Sosial*, 1973.

SISTEM KEAMANAN GANDA PADA SEPEDA MOTOR UNTUK PENCEGAHAN PENCURIAN DENGAN SMARTY(*SMART SECURITY*)

Dendy Pratama^{*}, Eko Didik Febriyanto¹, Denisson Arif Hakim¹, Tri Mulyadi, Umi Fadlilah¹, Raih Wisesa Alfiani²

¹Program Studi Teknik Elektro
Universitas Muhammadiyah Surakarta

²Program Studi Teknik Kimia
Universitas Muhammadiyah Surakarta

*d400130084@student.ums.ac.id

Abstrak

Alat pengaman sepeda motor adalah peralatan yang berfungsi mengamankan sepeda motor dari pencurian ketika tidak ada pengawasan dari pemiliknya. Alat pengaman sepeda motor terdiri dua macam: elektrik dan nonelektrik. Contoh pengaman sepeda motor elektrik yaitu menggunakan alarm, yang nonelektrik menggunakan kunci stir. Alat pengaman sepeda motor elektrik merupakan pengamanan yang efektif dalam melakukan fungsinya. Namun, alat pengaman tersebut belum bisa diandalkan karena jika pemilik sepeda motor berada jauh dari motornya, pemilik tidak dapat memantau kendaraannya. Masalah ini diatasi dengan pengembangan alat pengaman ganda. Alat pengaman berbasis Arduino ini dirancang untuk mengamankan sepeda motor secara efektif karena dilengkapi dengan GPS yang digunakan untuk melacak lokasi koordinat sepeda motor tanpa dibatasi jarak. Alat dilengkapi modul GSM sebagai komunikasi antara pemilik dengan alat pengaman sehingga pemilik dapat memantau sepeda motornya dari jarak jauh, yaitu menggunakan SMS yang berisi sandi. Bila sandi tersebut dikirimkan ke nomor *handphone* yang dipasang pada GSM modul maka alat bekerja sesuai dengan kata sandi. Hasil dari penelitian menunjukkan alat bekerja dengan baik. Percobaan pertama ketika kata sandi dikirim untuk menghidupkan kontak sepeda motor maka seketika kontak sepeda motor hidup. Selanjutnya kata sandi dikirim lewat SMS untuk mematikan mesin sepeda motor maka mesin tersebut mati. Ketika kata sandi dikirim untuk mengetahui lokasi alat yang terpasang pada sepeda motor tersebut, maka dalam 2 detik alat mengirimkan SMS berupa koordinat lokasi berbentuk *link* ke *google map*. Ketika sandi yang dikirim dalam bentuk SMS salah, alat tidak merespons. Alat ini menggunakan *keypad* yang digunakan untuk memasukkan data melalui *password* untuk membuka sistem keamanan dan hasilnya sesuai.

Kata Kunci: alat pengaman, Arduino, GPS, modul GSM, sepeda motor

1. Pendahuluan

Sepeda motor merupakan alat transportasi yang penting bagi kebanyakan masyarakat Indonesia karena harganya yang terjangkau bagi kebanyakan warga dengan penghasilan menengah ke bawah. Sepeda motor merupakan alternatif terbaik bagi banyak orang karena dapat berkelit dari kemacetan dan padatnya jalan raya Indonesia [1], [2]. Lubis menyatakan bahwa di tahun 2007 saja pertumbuhan sepeda motor telah melampaui 78% jumlah kendaraan di Indonesia atau lebih dari 37 juta unit dengan pertumbuhan 21% per tahun [3].

Pertambahan jumlah kendaraan bermotor seiring dengan pertambahan jumlah kejahatan yang melibatkan sepeda motor. Menurut Kapolda Jakarta Raya, selama kurun waktu 2015 di wilayah Jakarta, Depok, Tangerang, dan Bekasi (Jabodetabek) terdapat tujuh jenis tindak kriminal yang menjadi perhatian masyarakat, salah satu kasus yang menonjol adalah pencurian. Terjadi 101 pencurian kendaraan bermotor dari berbagai jenis sedangkan jumlah tersangka yang tertangkap hanya 12 orang [4].

Dengan banyaknya kejadian pencurian sepeda motor, maka sistem pengaman sepeda motor menjadi kebutuhan yang penting bagi pemilik sepeda motor. Sebenarnya telah tersedia berbagai sistem keamanan yang ditawarkan dan dapat digunakan oleh konsumen baik berupa pengaman nonelektrik maupun elektrik. Beberapa alternatif adalah kunci stang, kunci stang yang dipasang di cakram, kunci yang dipasang pada *hand* rem kanan yang dikaitkan dengan stang, kunci untuk menutup kontak, dan alarm yang menggunakan suara. Alarm suara menjadi indikator yang merupakan keamanan standar dari sebuah alarm. Alarm jenis ini membunyikan suara melalui pengeras suara (*speaker*) yang terpasang pada sepeda motor yang berfungsi untuk memberikan informasi kepada pemilik sepeda motor dan lingkungan sekitar bahwa kondisi motor tidak aman. Selain itu terdapat pengaman berupa gembok elektrik yang dipasang pada cakram sepeda motor di mana ketika kunci gembok tidak sesuai maka alarm yang berasal dari dalam gembok tersebut berbunyi sehingga memberikan informasi kepada masyarakat sekitar. Kelemahan dari pengaman yang ada seringkali hanya menggunakan satu protektor saja seperti

hanya menggunakan alarm sebagai alat pengamanannya ketika ada maling.

Hingga saat ini, masih kurangnya penggunaan *smartphone android* sebagai protektor perlindungan motor dari tindak kriminal. *Smartphone android* berfungsi sebagai pemantau ketika motor dicuri yang akan mengirim SMS.

Untuk mengatasi permasalahan ini, perlu dibuat sistem pengaman ganda yang dapat mencegah terjadinya pencurian, selain dari alarm yang merupakan suara dari pengeras suara sebagai indikator. Pengaman tambahan diperlukan misalnya berupa perangkat pengirim pesan yang memiliki jangkauan yang cukup luas sebagai pemberi informasi kepada pemilik sepeda motor dan membuat sepeda motor tidak dapat dioperasikan (dinyalakan).

Tulisan ini mendeskripsikan hasil penelitian untuk mengembangkan alat pengaman sepeda motor yang menggunakan teknik pada sistem alarm yang sudah ada dengan menghasilkan suara keras klakson sepeda motor ditambah dengan media informasi jarak jauh yang berupa *short message service* (SMS) sebagai pemberi pesan kepada pemilik sepeda motor bila terjadi pencurian. Sistem keamanan dibangun menggunakan Arduino sebagai alat untuk mengoperasikan sepeda motor. Sistem keamanan ini menggunakan teknik *engine stop* yang akan dihubungkan dan dihidupkan bersama dengan pengendali Arduino.

Pengendali Arduino selain digunakan untuk menghidupkan klakson sebagai alarm sepeda motor secara otomatis dan juga digunakan untuk mengirimkan pesan berupa SMS kepada pemilik sepeda motor melalui GSM modul yang dihubungkan dengan Arduino, sehingga pemilik dapat berkomunikasi dengan alat melalui *handphone*. Selain itu pengaman ini juga dilengkapi dengan komponen GPS yang berfungsi untuk memonitor sepeda motor tersebut. Jadi di mana saja letak sepeda motor akan diketahui oleh pemiliknya. Ketika motor berpindah lokasi maka pemilik akan mendapat titik koordinat di mana lokasi sepeda motor tersebut dipindahkan. Dengan pembuatan alat ini diharapkan meningkatkan tingkat keamanan sepeda motor.

2. Metode

2.1 Telaah Penelitian

Penggunaan modul GSM untuk sistem pemberi perintah dan monitoring jarak jauh dalam bentuk SMS telah dicontohkan sebelumnya dalam karya oleh [5]. Karya tersebut menceritakan bahwa sistem dapat memantau dan mengontrol rumah dari jarak jauh dan memberikan rasa aman kepada pengguna ketika berada jauh dari rumah. Sistem ini berfokus pada SMS dan menggunakan teknologi nirkabel seperti Wi-Fi. Contoh lain penggunaan modul GSM diterapkan oleh [6]. Keduanya menjelaskan bahwa pemanfaatan sistem dengan GSM dalam bentuk SMS yang digunakan sebagai pemantau sistem dari jarak jauh sangatlah efektif dan memiliki banyak kelebihan dan sangat memudahkan bagi penggunaannya.

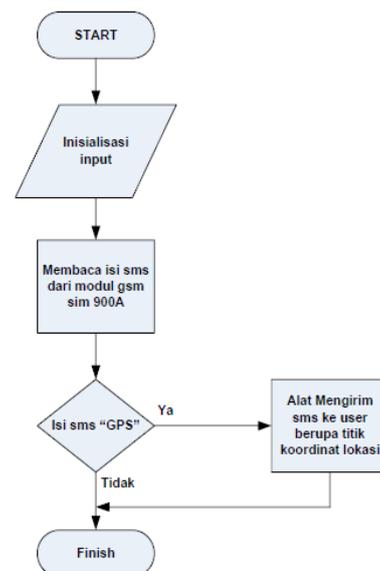
Penggunaan GPS dan GSM modul sebagai sistem keamanan telah diterapkan oleh Sandeep [7]. Karya ini menceritakan bahwa tentang banyaknya kecelakaan di mana korban kecelakaan tersebut meninggal karena

terlambat untuk dibawa ke rumah sakit. Kebanyakan orang yang ada di sekitar kecelakaan kurang tanggap untuk memanggil ambulans atau polisi dan juga kurang cepat untuk membawa korbannya ke rumah sakit. Latar belakang tersebut memberi ide kepada Sandeep untuk membuat helm cerdas yang dilengkapi dengan sensor getar, GPS, modul GSM, dan Arduino. Prinsip kerjanya yaitu ketika terjadi benturan yang keras pada helm maka sensor getar akan memberi masukan kepada Arduino kemudian Arduino memberi intruksi kepada GPS untuk memberikan lokasi. Selanjutnya data lokasi koordinat tersebut dikirim melalui modul GSM dalam bentuk SMS ke ambulans, kantor polisi, dan keluarga.

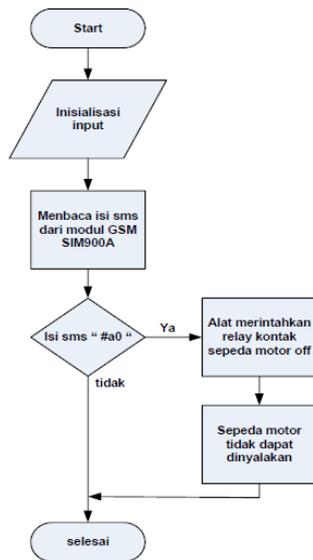
Perangkat ini juga memanfaatkan teknologi *bluetooth* untuk komunikasi jarak terbatas. Penggunaan teknologi semacam ini telah diterapkan oleh peneliti lain misalnya [8] untuk membuat novel berbasis audio bagi penyandang tuna netra.

2.2 Alur Penelitian

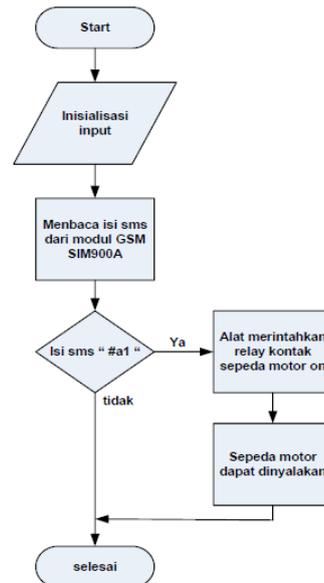
Penelitian ini mencakup empat proses yang dideskripsikan dalam 4 *flowchart*: yaitu *flowchart* koordinat titik/lokasi, *flowchart* kendali sistem alat, *flowchart* menghidupkan kontak sepeda motor menggunakan SMS, dan *flowchart* mematikan kontak sepeda motor menggunakan SMS. *Flowchart* koordinat titik lokasi yang ditampilkan pada Gambar 1 menjelaskan awal mula alat ini bekerja dengan menginisialisasi input yang dikemas dalam bentuk SMS dengan kode "g" dari user kemudian ketika SMS tersebut masuk maka GSM modul akan membaca isi dari SMS tersebut apakah sesuai dengan kode yang sudah dibangun di program atau kode tersebut tidak sesuai. Bila kode tersebut sesuai benar maka alat akan merespons dan mengirimkan data koordinat lokasi yang sudah dalam bentuk *link* yang dikirim lewat SMS ke user. Dan apabila isi SMS tidak sesuai dengan kode yang diminta maka alat tidak akan merespons.



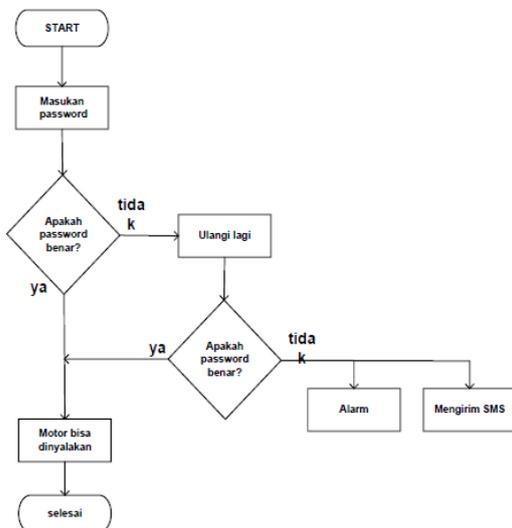
Gambar 1. *Flow Chart* koordinat titik lokasi



Gambar 2. Flowchart kendali sistem alat dengan Bluetooth



Gambar 4. Flowchart mematikan kontak sepeda motor menggunakan sms



Gambar 3. Flowchart menghidupkan kontak sepeda motor menggunakan sms

Flowchart kendali sistem alat seperti Gambar 2 dan Gambar 3 dengan bluetooth merupakan alur bagaimana sistem alat bekerja menggunakan bluetooth. Awal mula user harus memasukan password menggunakan bluetooth kemudian alat tersebut akan memeriksa apakah password yang dimasukan itu benar atau tidak. Apabila password tersebut benar maka alat akan merespons dan mengaktifkan saklar yang menghubungkan CDI (Capacitor Discharge Ignition) ke coil pada sepeda motor sehingga motor bisa dioperasikan. Apabila password tersebut tidak sesuai dengan kata kunci yang diminta maka alat akan memberi kesempatan kedua untuk mengulangi memasukan password. Apabila masih tidak sesuai dengan kata kunci yang diminta maka alarm sepeda motor akan aktif dan alat akan mengirim SMS warning kepada user.

Flowchart menghidupkan kontak sepeda motor seperti Gambar 3 menggunakan SMS merupakan alur bagaimana cara menggunakan SMS untuk menghidupkan kontak sepeda motor. Dimulai dengan user mengirim pesan SMS dengan kode "#a1" ke alat kemudian alat akan menginisialisasi SMS tersebut dan memeriksa apakah isi dari SMS sesuai dengan kode yang diminta atau tidak. Apabila isi SMS sesuai dengan kode yang diminta maka alat akan merespons dan mengaktifkan relay yang kemudian akan menghidupkan kontak sepeda motor dan sepeda motor dapat dinyalakan, tetapi apabila isi SMS tidak sesuai dengan kode yang diminta maka alat tidak merespons.

Flowchart mematikan kontak sepeda motor menggunakan SMS seperti Gambar 4 merupakan alur bagaimana cara menggunakan SMS untuk mematikan kontak sepeda motor. Dimulai dengan user mengirim pesan SMS kode "#a0" ke alat kemudian alat akan menginisialisasi SMS tersebut dan memeriksa apakah isi dari SMS sesuai dengan kode yang diminta atau tidak. Apabila isi SMS sesuai dengan kode yang diminta maka alat akan merespons dan mengaktifkan relay yang kemudian akan mematikan kontak sepeda motor. Apabila isi SMS tidak sesuai dengan kode yang diminta maka alat tidak merespons.

3. Hasil dan Pembahasan

3.1 Hasil Pengembangan Alat

Bentuk tampilan fisik dari perancangan alat ini dapat dilihat pada Gambar 5-17. Gambar 5 merupakan tampilan fisik di dalam box dan Gambar 6 tampilan fisik ketika box ditutup yang mana terdapat 5 led dan sebuah keypad.

Pengujian membuktikan bahwa ketika saklar untuk menghidupkan alat diaktifkan maka indikator led warna putih menyala yang artinya alat ini sudah aktif. Tampilan

indikator nyala led warna putih dapat dilihat pada Gambar 7. Pada langkah selanjutnya jika *user* ingin mengaktifkan *relay* yang digunakan untuk menghubungkan CDI ke *coil* agar kontak sepeda motor dapat menyala maka ada 2 cara. Cara yang pertama menggunakan *keypad* untuk memasukkan *password* yang digunakan untuk membuka sistem keamanan. Apabila *password* salah maka alat mengirimkan pesan *warning* kepada pemilik sepeda motor kemudian lampu warna merah menyala dan bersamaan dengan itu klakson sepeda motor akan berbunyi. Hasil dapat dilihat pada Gambar 8 dan untuk pesan *warning* dari alat dapat dilihat pada Gambar 9 dengan cara mengirimkan SMS dengan format "g" ke nomor yang dipasang pada *gsm* modul.

Hasilnya dapat dilihat pada tampilan Gambar 9 dan 10, di mana SMS balasan dari alat led. Apabila *password* yang dimasukkan menggunakan *keypad* sesuai dengan kata kunci untuk membuka sistem maka led warna hijau sebagai *indicator* sistem on kontak akan menyala dan sepeda motor bisa dihidupkan. Hasilnya dapat dilihat pada Gambar 10.



Gambar 5. Tampilan alat di dalam *box*



Gambar 6. Tampilan alat dari luar *box*



Gambar 7. Tampilan alat dengan *indicator* led putih menyala saat saklar diaktifkan



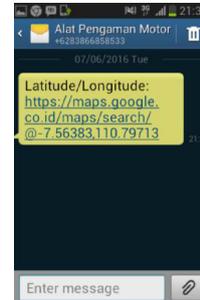
Gambar 8. Tampilan alat dengan *indicator* led merah menyala saat saklar diaktifkan



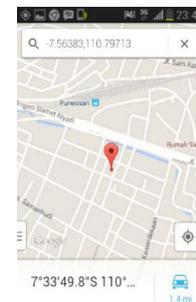
Gambar 9. Tampilan pesan *warning* dari alat ketika *password* yang dimasukkan salah



Gambar 10. Tampilan Alat dengan *indicator* led hijau menyala ketika *password* yang dimasukkan benar



Gambar 11. Tampilan pesan balasan sms dari alat yang berisi *link* lokasi keberadaan alat



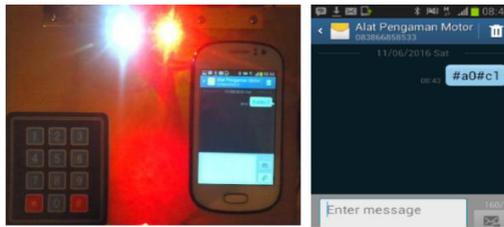
Gambar 12. Tampilan koordinat melalui *google maps*

Hasil berikutnya yaitu mencari lokasi titik koordinat dari GPS yang dipasang pada sepeda motor yang dikirim ke *user* adalah berupa data *latitude* dan *longitude*, akan tetapi data *latitude* dan *longitude* tersebut akan di-*parsing* terlebih dahulu sehingga akan diterima data tersebut berupa *link* yang langsung terhubung dengan *Google maps*.

Hasil dari penelitian selanjutnya adalah membuka sistem keamanan melalui SMS agar kontak dapat menyala dan sepeda motor dapat dinyalakan. Langkah untuk membuka sistem melalui SMS adalah dengan cara mengirim pesan SMS ke nomor alat yang dipasang pada *gsm* modul dengan format #a1 dan hasilnya sesuai yang diharapkan yaitu sistem dapat dibuka yang ditandai dengan led hijau menyala sesuai dengan tampilan pada Gambar 13. Kemudian untuk mematikan mesin sepeda motor dan mengaktifkan alarm dari jarak jauh dengan bantuan SMS pula (Gambar 14).



Gambar 13. Tampilan alat ketika sistem keamanan dibuka



Gambar 14. Tampilan alat ketika mesin motor dimatikan dan alarm diaktifkan melalui sms

```

if (inchar=='#')
{
  delay(10);
  inchar=SIM900.read();
  if (inchar=='a')
  {
    delay(10);
    inchar=SIM900.read();
    if (inchar=='0')
    {
      digitalWrite(kontak, LOW);
    }
    else if (inchar=='1')
    {
      digitalWrite(kontak, HIGH);
    }
    delay(10);
  }
}

```

Gambar 15. Contoh kode program menyalakan dan mematikan melalui kontak dengan sms

```

{
  SIM900.print("AT+CMGF=1\r"); // AT command to send SMS me
  delay(100);
  SIM900.println("AT + CMGS = \""+6283866074853+"\"); //recip
  delay(100);
  SIM900.print("Latitude/Longitude: https://maps.google.co.
  SIM900.print(flat,5); // message to send
  SIM900.print(","); // message to send
  SIM900.println(flon,5); // message to send
  delay(100);
  SIM900.println((char)26); // End AT command with a ^Z, A
  delay(100);
  SIM900.println();
  delay(5000);
  SIM900power(); // turn off module
}

```

Gambar 16. Contoh kode program mengirim data *longitude* dan *latitude* yang dikirim lewat sms



Gambar 17. Contoh pemasangan alat pada sepeda motor "Vega"; Gambar kiri atas adalah pemasangan kabel sumber tegangan pada aki sepeda motor; Gambar tengah dan kanan atas adalah pemasangan kabel *relay* pada kunci kontak; Gambar kiri dan tengah bawah adalah pemasangan kabel *relay alarm*; Gambar tengah dan kanan bawah memperlihatkan posisi alat pada sepeda motor.

Tabel 1. Hasil pengujian fungsional

Skenario Pengujian	Hasil yang diharapkan	Pengamatan	Makna
Mengaktifkan saklar	Led putih nyala sebagai tanda alat sudah aktif	Led putih nyala dan alat keamanan aktif	Berhasil
Memasukkan <i>password</i> melalui <i>keypad</i> untuk membuka sistem keamanan.	Alat mengeksekusi perintah dan led hijau menyala sebagai tanda motor bisa dihidupkan.	Alat merespons dan menyalakan led hijau tanda sepeda motor bisa dihidupkan.	Berhasil
Memasukkan <i>password</i> yang salah sejumlah 2 kali.	Alat mengirimkan pesan <i>warning</i> ke <i>user</i> selanjutnya alat mengaktifkan alarm dan led merah menyala.	Alat merespons dan mengirimkan pesan <i>warning</i> ke <i>user</i> kemudian alarm dan led merah aktif.	Berhasil
<i>User</i> mengirim sms ke alat dengan kode "g"	Alat merespons perintah dan mengirimkan data lokasi koordinat posisi sepeda motor kepada <i>user</i> melalui sms.	Alat mengeksekusi perintah dan mengirimkan data lokasi berupa data <i>latitude</i> dan <i>longitude</i> yang sudah di- <i>parsing</i> ke <i>link google maps</i> .	Berhasil
<i>User</i> mengirim sms ke alat dengan kode "#a1"	Alat merespons perintah dan kemudian mengeksekusi perintah tersebut	Alat merespons dan membuka sistem keamanan sehingga sepeda motor dapat dinyalakan.	Berhasil
<i>User</i> mengirim sms ke alat dengan kode "#a0#c1"	Alat merespons perintah dan kemudian mengeksekusi perintah tersebut	Alat merespons dan bekerja untuk mematikan mesin sepeda motor dan mengaktifkan alarm.	Berhasil

Tabel 2. Data pengujian keakuratan deteksi koordinat lokasi

Lokasi	Titik koordinat	
	latitude	longitude
	-7.56383	110.79713
Tegalrejo RT 01,	-7.56379	110.79713
RW 02, Sondakan,	-7.56383	110.79713
Laweyan, Surakarta	-7.56383	110.79713
	-7.56383	110.79713

Sehingga apabila sistem keamanan motor tersebut dapat dibuka kemudian sepeda motor dapat dinyalakan maka saat sepeda motor tersebut digunakan oleh pencuri tadi, pemilik dapat mematikan sepeda motor dan mengaktifkan *alarm* dari jarak jauh dan pencurian dapat digalkan. Format dari SMS ini adalah "#a0#c1".

Gambar 15 memperlihatkan kode program untuk menyalakan dan mematikan kontak melalui sms. Sedangkan Gambar 16 memperlihatkan program untuk mengirim data *longitude* dan *latitude* melalui sms.

Gambar 17 memperlihatkan foto-foto pemasangan alat pada sebuah sepeda motor. Modifikasi diperlukan pada sepeda motor untuk mengendalikan kabel kontak agar dapat diputus dan dihubungkan oleh alat. Sumber daya bagi alat diperoleh dari aki sepeda motor. Selebihnya diperlukan tempat (*space*) untuk menempatkan alat pada sepeda motor. Jika dimungkinkan alat ditempatkan pada bagasi sehingga tidak tampak.

3.2 Pengujian

Pengujian alat dilakukan dalam dua tahap. Tahap pertama adalah pengujian fungsional, yaitu menguji berbagai skenario fungsi apakah telah bekerja sesuai yang diharapkan. Kedua adalah pengujian deteksi lokasi dan pengiriman lokasi kendaraan. Pengujian kedua mensimulasikan kendaraan dalam status sudah dicuri untuk melihat apakah alat mampu mengirim lokasi koordinat dan merespons perintah secara jarak jauh.

Hasil pengujian fungsional disajikan dalam Tabel 1. Ada enam skenario yang diuji yaitu fungsi mengaktifkan saklar, memasukkan *password* untuk akses keamanan, memasukkan *password* salah, mengirim perintah dengan kode "g", kode "#a1" dan kode "#a0#c1". Tabel 1 menunjukkan bahwa semua skenario berfungsi sesuai rancangan.

Pengujian kedua adalah deteksi lokasi dan pengiriman data lokasi. Deteksi lokasi oleh GPS dimaksudkan untuk mendapatkan koordinat titik lokasi sepeda motor yang telah dipasang alat pengaman ini. Pengujian mengambil satu sampel lokasi dan sampel dilakukan sebanyak 5 kali. Pada setiap sampel lokasi, dilakukan pengujian deteksi lokasi dalam dua keadaan motor, yaitu motor diam (Tabel 2) dan motor bergerak dengan kecepatan sekitar 15 km/jam (Tabel 3).

Tabel 2 memperlihatkan data hasil deteksi lokasi yang dikirim oleh alat sebanyak 5 kali dari lokasi yang sama, sebuah titik di Tegalrejo RT 01 RW 02 Sondakan, Laweyan, Surakarta. Alat menghasilkan deteksi nilai *longitude* yang sama untuk lima kali percobaan. Deteksi nilai *latitude* tidak berbeda signifikan di antara lima hasil percobaan, dan boleh dianggap sama.

Tabel 3. Data pengujian keakuratan deteksi koordinat saat sepeda motor bergerak

Lokasi	Acuan		Data pengujian	
	Waktu	Koordinat	Waktu	Koordinat
Jalan Parangkusumo, Sondakan, Laweyan, Surakarta			3"	7.563458, 110.795324
			3"	7.563462, 110.795335
	2"	7.563436, 110.795259	4"	7.563476, 110.795380
			2"	7.563449, 110.795319
			3"	7.563456, 110.795226

Tabel 3 memperlihatkan data pengujian deteksi lokasi ketika kendaraan bergerak. Data diambil sebanyak lima kali dan berlokasi di Jalan Parangkusumo, Sondakan, Laweyan, Surakarta. Kecepatan kendaraan adalah 15 km/jam. Selain lokasi koordinat, tabel juga menyajikan rentang waktu yang diperlukan sejak alat distimulus untuk mengirim SMS hingga pesan singkat tersebut diterima oleh *smartphone* tujuan. Sebagai pembanding, disajikan pula hasil deteksi koordinat untuk lokasi yang sama ketika dikirim oleh alat dalam keadaan diam. Hasil pengujian menunjukkan adanya selisih baik dari segi waktu pengiriman maupun koordinat yang dikirim oleh alat. Selisih ini relatif kecil. Misalnya untuk hasil pengujian pertama yaitu koordinat 7.563458, 110.795324 hanya berjarak delapan meter dari titik acuan yaitu koordinat 7.563436, 110.795259. Dalam radius delapan meter, memang bisa terdapat beberapa sepeda motor, namun sepeda motor yang dicari dapat diharapkan mudah dikenali.

Waktu pengiriman koordinat saat kendaraan bergerak berbeda cukup signifikan dari waktu saat kendaraan diam. Untuk kecepatan 15 km/jam, waktu pengiriman selama 3 detik setara dengan gerakan kendaraan sejauh 12 meter. Jika kendaraan yang dicuri dalam keadaan bergerak dan kita berada di lokasi pengiriman tiga detik yang lalu, maka kemungkinan kendaraan telah bergerak 12 meter.

4. Kesimpulan

Berdasarkan hasil pengujian di atas dapat ditarik kesimpulan bahwa alat pengaman sepeda motor dengan sistem kontrol Arduino dapat bekerja dengan baik sesuai dengan rancangannya. Alat ini dapat dikontrol dari jarak dekat maupun jarak yang jauh dikarenakan menggunakan sistem SMS dan GPS satelit. Alat dapat merespons dengan cepat ketika ada instruksi yang masuk sesuai dengan rancangannya.

Selanjutnya perlu dilakukan pengujian ketika kendaraan bergerak dengan cepat misalnya pada kecepatan 40 km/jam dan 80 km/jam, yaitu kecepatan normal kendaraan di kota dan kecepatan kendaraan yang sedang negebut karena pengendara panik sedang melarikan diri.

5. Daftar Pustaka

- Siska Febri, "Studi Deskriptif Mengenai Motivasi Melakukan Tingkah Laku Berkendara Beresiko pada Pengendara Sepeda Motor di Jatinangor," Universitas Padjadjaran, Bandung, Skripsi 2016.
- Amelia Kusuma Indriastuti and Harmen Sulistio, "Influencing Factors on Motorcycle Accident in Urban Area of Malang, Indonesia," *International Journal of Academic Research*, vol. 2, no. 5, pp. 252-255, 2010.
- H. A. R. S. Lubis, "The Increase of Motorcycles and its Impact on Urban Transportation," *Jurnal Transportasi*, vol. 8, no. 3, p. 187, 2008.
- Gardi Gazarin. (2016, Januari) Tahun 2015, Tingkat Kejahatan di Jadetabek Tinggi. [Online]. HYPERLINK "<http://www.beritasatu.com/hukum-kriminalitas/338558-tahun-2015-tingkat-kejahatan-di-jadetabek-tinggi.html>" <http://www.beritasatu.com/hukum-kriminalitas/338558-tahun-2015-tingkat-kejahatan-di-jadetabek-tinggi.html>
- Minal S. Khandare and Anjali Mahajan, "Mobile Monitoring System for Smart Home," in *International Conference on Emerging Trends in Engineering*, 2010, pp. 848-852.
- Chen Peijian and Jiang Xuehua, "Design and Implementation of Remote Monitoring System Based on GSM," in *IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, 2008, pp. 678-681.
- Sandeep Baghel, "Smart Helmet," *IEEE Transmitter*, p. 49, 2011.
- Dendy Pratama et al., "Rancang Bangun Alat dan Aplikasi untuk Para Penyandang Tunanetra Berbasis Smartphone Android," *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, vol. 2, no. 1, pp. 14-19, 2016.

PENERAPAN FRAMEWORK COBIT UNTUK IDENTIFIKASI TINGKAT KEMATANGAN TATA KELOLA TEKNOLOGI INFORMASI: STUDI KASUS DI FASILKOM UNWIDHA

Agustinus Suradi^{1*}, Sri Wiyanta¹

¹Universitas Widya Dharma

*simpati2000@mailcity.com

Abstrak

Perkembangan IT sebagai media komunikasi data hingga saat ini berkembang sangat pesat. Peranan teknologi informasi dalam suatu lembaga sangat krusial, namun masalah yang sering terjadi di lembaga/ instansi adalah penggunaan teknologi informasi yang ternyata tidak sesuai dengan harapan, oleh karena itu diperlukan tata kelola terhadap penggunaan teknologi informasi. Permasalahan yang dihadapi di Fasilkom Unwidha adalah adanya ketidakpuasan layanan IT yang diberikan, terdapatnya gangguan layanan yang tidak segera diatasi, adanya koordinasi maintenance infrastruktur IT yang kurang optimal. Sehingga perlu adanya evaluasi terhadap tata kelola teknologi informasi yang diberikan untuk penyesuaian dengan tujuan organisasi agar visi yang diharapkan dapat tercapai. Penelitian ini membahas tentang kondisi tata kelola teknologi informasi di Fasilkom Unwidha, dan sejauh mana telah menerapkan tata kelola teknologi informasi yang baik. Metode analisis yang digunakan dalam penelitian ini adalah dengan menggunakan framework Control Objectives for Information and Related Technology (COBIT). Hasil dari analisis ini berupa tingkat kematangan tata kelola teknologi informasi yang mencerminkan kondisi tata kelola teknologi informasi di Fakultas Ilmu Komputer Unwidha. Tingkat kematangan dengan mengacu pada maturity level yang disediakan kerangka kerja COBIT, diperoleh nilai kematangan 2,84 yaitu pada level Defined Process. Sistem memiliki ruang lingkup Business Goals sebanyak 13, IT Goals sebanyak 18, IT Process sebanyak 30 proses dan 4 Domain.

Kata Kunci: *IT Governance, maturity level, COBIT.*

1. Pendahuluan

Pengelolaan sistem informasi yang tepat akan berdampak pada suatu instansi dalam mencapai target tujuannya. Pesatnya perkembangan ilmu pengetahuan dan teknologi khususnya di bidang IT sangat berperan dalam mendukung kemajuan suatu organisasi.

Teknologi Informasi saat ini menjadi bagian yang tak terpisahkan dan terintegrasi dengan tujuan bisnis organisasi. Bagaimana teknologi informasi diaplikasikan dalam suatu organisasi akan mempengaruhi seberapa jauh organisasi tersebut telah mencapai visi, misi, ataupun tujuan strategisnya [1].

Agar implementasi *IT Governance* pada Fakultas dapat berlangsung secara efektif, organisasi harus dapat mengevaluasi dan menilai sejauh mana *IT Governance* yang sekarang berlangsung dan mengidentifikasi peningkatan yang dapat dilakukan. Hal tersebut berlaku pada semua proses yang dikelola yang terkandung dalam TI dan proses *IT Governance* itu sendiri. Pada penelitian [2] penggunaan model *maturity* dalam hal ini akan memudahkan dalam penilaian dengan cara pendekatan yang terstruktur terhadap skala yang mudah dimengerti dan konsisten.

Berkaitan dengan *IT Governance*, diperlukan adanya sebuah mekanisme audit terhadap pengelolaan teknologi informasi. Secara umum kerangka kerja tata kelola TI serta pengendalian yang dibutuhkan untuk mencapainya

disediakan oleh *framework* COBIT (*Control Objective for Information and related Technology*) [3], [4].

Pada penelitian ini bertujuan untuk menganalisis tata kelola IT dan melakukan pengukuran *maturity level* tata kelola IT di Fasilkom Unwidha agar memberikan pelayanan yang maksimal dan didukung oleh pengelolaan manajemen yang handal sehingga dapat tercapai tujuan organisasi seperti yang diharapkan.

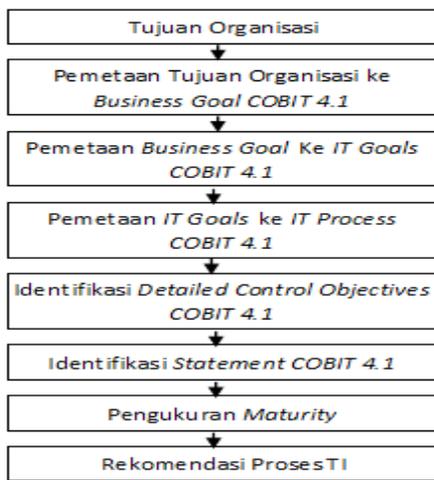
2. Metode Penelitian

Metode penelitian yang digunakan yaitu metode kualitatif, prosedur penelitian yang menghasilkan data deskriptif. Dalam penelitian ini akan digunakan studi kasus Fakultas Ilmu Komputer yang bertipe deskriptif dengan tujuan agar peneliti mendapatkan gambaran yang sejelas mungkin mengenai proses tata kelola TI, terutama yang berkaitan dengan efektivitas dan efisiensi.

2.1 Tahapan Penelitian

Metode analisis data yang dilakukan dengan menggunakan kerangka kerja yang terdapat dalam Cobit 4.1. Berikut adalah beberapa tahapan yang akan dilakukan dalam analisis data [5]:

- Tahap 1 – *Gain Understanding*.
- Tahap 2 – *Study Literatur*.



Gambar 1. Alur penelitian

Tabel 1. IT Process dan IT Domain yang Teridentifikasi

IT PROCESS	IT DOMAIN
PO1, PO2, PO3, PO4, PO5, PO6, PO7, PO8, PO9, PO10	Plan and Organise
AI1, AI2, AI3, AI4, AI5, AI6, AI7	Acquire and Implementation
DS1, DS2, DS3, DS4, DS5, DS6, DS7, DS8, DS10, DS12, DS13	Deliver and Support
ME1, ME4	Monitor and Evaluation

- Tahap 3 – Desain Penelitian
- Tahap 4 – Pengolahan Data
- Tahap 5 – Tahap Analisis Data
- Tahap 6 – Penyusunan Laporan

2.2 Teknik Pengumpulan Data

Dalam COBIT terdapat panduan mengenai kegiatan yang penting untuk dilakukan berkaitan dengan proses tata kelola TI. Oleh karena itu, alat pengumpulan data dikembangkan berdasarkan indikator-indikator kegiatan yang terdapat pada *framework* COBIT. Data utama dikumpulkan dengan kuesioner dan dilengkapi dengan wawancara, observasi, serta kepustakaan dan dokumen tertulis.

- a. Wawancara

Kegiatan ini dilakukan untuk menangkap deskripsi lebih lengkap mengenai masalah yang diteliti yang tidak terjaring melalui kuesioner. Untuk pedoman wawancara, meneliti juga berpedoman pada model kematangan dan *Control Objectives* COBIT.
- b. Observasi

Pengamatan dilakukan terhadap pengelolaan TI berdasarkan aspek-aspek yang telah ditetapkan dalam COBIT.
- c. Kuesioner

Untuk kuesioner mengenai tingkat kesadaran pengelolaan mengenai pengelolaan, proses, pengawasan, dan evaluasi TI, peneliti menggunakan COBIT 4.1, yaitu mendata kegiatan-kegiatan yang

berhubungan dengan tata kelola TI, apa saja yang dinilai penting menurut COBIT, dan menanyakan tingkat keperluan kegiatan-kegiatan tersebut kepada responden [6]

2.3 Alur Penelitian

Berdasarkan tahapan penelitian yang akan dilakukan maka dibuatlah alur penelitian dari pengukuran Maturity Model [7], [8], [9]:

3. Hasil Penelitian Dan Pembahasan

Analisis Tata Kelola Teknologi Informasi Dengan Menggunakan COBIT 4.1 Maturity Model dengan:

- a. Pemetaan Tujuan organisasi ke *Business Goals* COBIT 4.1
 - a) Pemetaan ke *COBIT 4.1 IT Goals*.
 - b) Pemetaan ke *COBIT 4.1 IT Process*.
 - c) Pemetaan ke *Detailed Control Objectives*.
 - d) Dari *Detailed Control Objectives* ke *Statements* dalam setiap proses TI
 - e) Dari *Statements* dari setiap tingkat dalam setiap proses dinilai berdasarkan cara Pederiva [10].
 - f) Penilaian tingkat kematangan setiap Proses TI.
 - g) Penilaian tingkat kematangan setiap Domain TI.
 - h) Penilaian tingkat kematangan total dari setiap Domain TI.
- b. Identifikasi Business Goals COBIT 4.1

Mengidentifikasi tujuan (*goals*) organisasi selanjutnya dilakukan pemetaan dari tujuan organisasi tersebut dengan *business goals* yang dimiliki COBIT 4.1. Proses pemetaan dilakukan dengan menghubungkan *business goals* pada COBIT 4.1
- c. Identifikasi IT Goals COBIT 4.1

Setelah mengidentifikasi Tujuan Bisnis, langkah selanjutnya adalah mengidentifikasi Tujuan TI yang sesuai dengan studi kasus Fakultas. COBIT telah memetakan *Business Goals* dengan *IT Goals*, dan dari pemetaan tersebut terlihat *IT Goals* apa saja yang nantinya akan menunjang *Business Goals* Fakultas.
- d. Identifikasi IT Process COBIT 4.1

Setiap tujuan teknologi informasi dapat terdiri atas beberapa proses teknologi informasi yang terkait, demikian juga sebaliknya setiap *IT Process* dapat digunakan untuk memenuhi beberapa *IT Goals*. Hasil pemetaan antara *IT Goals* dan *IT Process* dalam kerangka kerja COBIT 4.1

Pemetaan *IT Process* yang dapat diterapkan menghasilkan 30 (tiga puluh) proses. Adapun *IT Process* yang sesuai dengan *IT Goals* sebagai berikut:

 - PO1: Define a Strategic Plan
 - PO2: Define the Information Architecture
 - PO3: Determine Technological Direction
 - PO4: Define the IT Processes, Organization and Relationship
 - PO5: Manage the IT Investment
 - PO6: Communicate Management Aims and Direction
 - PO7: Manage IT Human Resources
 - PO8: Manage Quality
 - PO9: Asses and Manage IT Risks
 - PO10: Manage Projects
 - AI1: Identify Automated Solution
 - AI2: Acquire and Maintain Application Software

- AI3: Acquire and Maintain Technology Infrastructure
 - AI4: Enable Operation and Use
 - AI5: Procure IT Resources
 - AI6: Manage Changes
 - AI7: Install and Accredite Solutions and Changes
 - DS1: Define and manage service levels
 - DS2: Manage third-party services
 - DS3: Manage performance and capacity
 - DS4: Ensure continuous service
 - DS5: Ensure systems security
 - DS6: Identify and allocate costs
 - DS7: Educate and train users
 - DS8: Manage service desk and incidents
 - DS10: Manage problems
 - DS12: Manage the physical environment
 - DS13: Manage operations
 - ME1: Monitor and Evaluate IT Performance
 - ME4: Provide IT Governance
- e. Tingkat Kematangan Tata Kelola TI
Hasil pemetaan *IT Process* di Fasilkom dengan COBIT 4.1 menghasilkan 30 Proses TI. Proses tersebut kemudian dinilai menggunakan COBIT 4.1 *Maturity Model*. Tabel berikut merupakan hasil penilaian terhadap *IT Process*:

Tabel 2. Tingkat Kematangan *IT Process* Domain PO

MATURITY LEVEL			
Domain		Plan and Organise (PO)	
No.	Kode	Proses TI	Nilai
1	PO1	Define a Strategic Plan	2,71
2	PO2	Define the Information Architecture	2,97
3	PO3	Determine Technological Direction	2,94
4	PO4	Define the IT Processes, Organization and Relationship	2,80
5	PO5	Manage the IT Investment	2,93
6	PO6	Communicate Management Aims and Direction	2,34
7	PO7	Manage IT Human Resources	2,82
8	PO8	Manage Quality	2,46
9	PO9	Asses and Manage IT Risks	2,71
10	PO10	Manage Projects	2,42
Nilai Total			27,09
Nilai kematangan Plan and Organise (PO)			2,71

Tabel 3. Tingkat Kematangan *IT Process* Domain AI

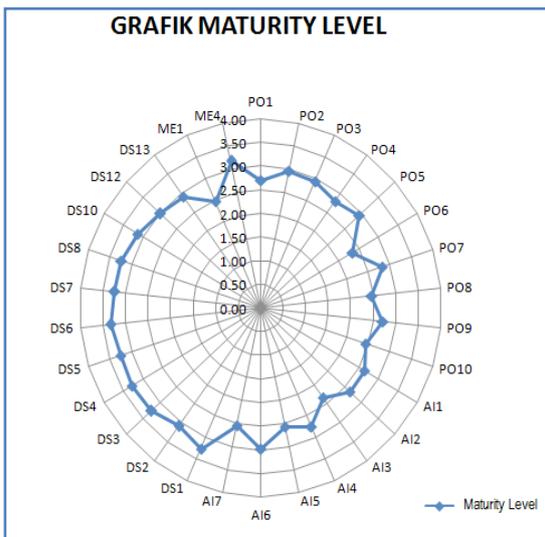
MATURITY LEVEL			
Domain		Acquire and Implement (AI)	
No.	Kode	Proses TI	Nilai
1	AI1	Identify Automated Solution	2,65
2	AI2	Acquire and Maintain Application Software	2,64
3	AI3	Acquire and Maintain Technology Infrastructure	2,34
4	AI4	Enable Operation and Use	2,74
5	AI5	Procure IT Resources	2,58
6	AI6	Manage Changes	2,98
7	AI7	Install and Accredite Solutions and Changes	2,55
Nilai Total			18,48
Nilai kematangan Domain Acquire and Implement (AI)			2,64

Tabel 4. Tingkat Kematangan *IT Process* Domain DS

MATURITY LEVEL			
Domain		Deliver and Support (DS)	
No.	Kode	Proses TI	Nilai
1	DS1	Define and manage service levels	3,26
2	DS2	Manage third-party services	3,09
3	DS3	Manage performance and capacity	3,26
4	DS4	Ensure continuous service	3,30
5	DS5	Ensure systems security	3,25
6	DS6	Identify and allocate costs	3,35
7	DS7	Educate and train users	3,25
8	DS8	Manage service desk and incidents	3,24
9	DS10	Manage problems	3,14
10	DS12	Manage the physical environment	2,99
11	DS13	Manage operations	2,91
Nilai Total			35,04
Nilai kematangan Deliver and Support (DS)			3,19

Tabel 5. Tingkat Kematangan IT Process Domain ME

MATURITY LEVEL			
Domain		Monitor and Evaluate (ME)	
No.	Kode	Proses TI	Nilai
1	ME1	Monitor and Evaluate IT Performance	2,47
2	ME4	Provide IT Governance	3,18
Nilai Total			5,65
Nilai kematangan Monitor and Evaluate (ME)			2,83



Gambar 2. Radar Chart Nilai kematangan IT Process Fasilkom

Tabel 6. Tingkat Kematangan Total IT Process

COBIT 4.1 MATURITY LEVEL	
Domain	Nilai
Plan and Organise (PO)	2,71
Acquire and Implement (AI)	2,64
Deliver and Support (DS)	3,19
Monitor and Evaluate (ME)	2,83
Nilai Total	11,36
Nilai kematangan Total IT Process	2,84

Dari penilaian tersebut dapat dikelompokkan, yaitu: Kategori pertama merupakan proses pada *Repeatable but intuitive* terdiri atas 5 proses, yaitu:

- PO6 *Communicate management aims and direction*, nilai 2,34
- PO8 *Manage quality human resource*, nilai 2,46
- PO10 *Manage projects*, nilai 2,42
- AI3 *Acquire and maintain technology infrastructure*, nilai 2,34
- ME1 *Monitor and evaluate IT performance*, nilai 2,47

Kategori kedua merupakan proses pada tahap *Defined Process* yang terdiri atas 25 proses, yaitu:

- PO1 *Define a strategic IT plan*, nilai 2,71
- PO2 *Define the information architecture*, nilai 2,97
- PO3 *Determine technological direction*, nilai 2,94
- PO4 *Define the IT processes, organization and relationships*, nilai 2,80
- PO5 *Manage the IT investment*, nilai 2,93
- PO7 *Manage IT human resources*, nilai 2,82
- PO9 *Asses and manage IT risks*, nilai 2,71
- AI1 *Identify automated solutions*, nilai 2,65
- AI2 *Acquire and maintain application software*, nilai 2,64
- AI4 *Enable operation and use*, nilai 2,74
- AI5 *Procure IT resources*, nilai 2,58
- AI6 *Manage changes*, nilai 2,98
- AI7 *Install and accredit solutions and changes*, nilai 2,55
- DS1 *Define and manage service levels*, nilai 3,26
- DS2 *Manage third-party services*, nilai 3,09
- DS3 *Manage performance and capacity*, nilai 3,26
- DS4 *Ensure continuous service*, nilai 3,30
- DS5 *Ensure systems security*, nilai 3,25
- DS6 *Identify and allocate costs*, nilai 3,35
- DS7 *Educate and train users*, nilai 3,25
- DS8 *Manage service desk and incidents*, nilai 3,24
- DS10 *Manage problems*, nilai 3,14
- DS12 *Manage the physical environment*, nilai 2,99
- DS13 *Manage operations*, nilai 2,91
- ME4 *Provide IT Governance*, nilai 3,18

Dari hasil nilai kematangan proses TI di atas dapat diperoleh nilai kematangan total proses TI, sebagai berikut:

Sehingga dari hasil penilaian diperoleh hasil tingkat kematangan proses tata kelola TI dengan nilai total kematangan **2,84** yaitu pada level **Define Process**, artinya sudah ada prosedur yang memiliki standar dan didokumentasikan dengan baik, sudah ada pelatihan formal untuk mengkomunikasikan prosedur dan kebijakan yang dibuat. Tetapi pada tahap implementasinya masih tergantung pada individu apakah mau melakukan prosedur yang ditetapkan atau tidak. Prosedur yang dibuat masih terbatas pada bentuk formalisasi dari praktik yang ada.

f. Analisis Rekomendasi Tata Kelola TI

Pada proses TI PO6 (*Communicate management aims and direction*), PO8 (*Manage quality human resource*), PO10 (*Manage projects*), AI3 (*Acquire and maintain technology infrastructure*), ME1 (*Monitor and evaluate IT performance*), pada level 2 (*Repeatable but Intuitive*), maka perlu adanya perbaikan untuk dapat ditingkatkan kematangannya. Agar proses PO6, PO8, PO10, AI3, dan ME1 dapat mencapai *maturity* level 3, maka yang perlu dilakukan adalah:

- PO6 (*Communicate management aims and direction*):
 - Kerangka kerja, kebijakan, perencanaan, dan prosedur dikembangkan, didokumentasikan, dan dikomunikasikan oleh pihak manajemen.
 - Proses pengembangan kebijakan yang terstruktur, dipelihara, dan mendapat dukungan oleh staff
 - Adanya teknik untuk mempromosikan

- kesadaran keamanan TI telah dibakukan dan diformalkan
- Ada kerangka kerja pengembangan global.
 - PO8 (*Manage quality human resource*):
 - Sebuah proses *Quality Management System* dikomunikasikan ke semua unsur oleh manajemen dan melibatkan TI dan pengguna akhir manajemen.
 - Adanya sebuah program pendidikan dan pelatihan untuk mengajarkan ke semua tingkat organisasi tentang kualitas.
 - Adanya alat umum dan praktik manajemen mutu yang muncul.
 - Kualitas survei kepuasan direncanakan dan kadang-kadang dilakukan.
 - PO10 (*Manage projects*):
 - Proses dan metodologi manajemen proyek TI perlu ditetapkan dan dikomunikasikan.
 - Senior TI dan pihak manajemen mulai berkomitmen dan terlibat dalam pengelolaan proyek-proyek TI.
 - Proyek TI yang ditugaskan perlu ada target bisnis dan teknis
 - Proyek TI dipantau anggaran dan pengukuran kinerjanya
 - AI3 (*Acquire and maintain technology infrastructure*):
 - Perlu pendefinisian yang jelas untuk mendapatkan dan memelihara infrastruktur TI.
 - Pemeliharaan infrastruktur TI direncanakan dijadwalkan dan dikoordinasikan.
 - ME1 (*Monitor and evaluate IT performance*):
 - Alat untuk memantau proses dan layanan didefinisikan.
 - Adanya pengukuran kinerja TI, pengukuran strategis, dan pengukuran kepuasan pelanggan.
 - Menetapkan sebuah kerangka kerja untuk mengukur kinerja.

4. Kesimpulan

Hasil penelitian ini menunjukkan pengukuran tingkat kematangan proses tata kelola TI di Fasilkom Unwidha dengan COBIT 4.1 maturity model memiliki ruang lingkup *Business Goals* sebanyak 13, *IT Goals* sebanyak 18, *IT Process* sebanyak 30 proses, dan 4 Domain.

Untuk analisis kematangan tata kelola TI, dapat dikelompokkan menjadi 2 yaitu: kategori pertama merupakan proses pada *level Repeatable but Intuitive* terdiri atas 5 proses, yaitu: PO6, PO8, PO10, AI3, ME1, dan kategori kedua merupakan proses pada *level Defined Process* yang terdiri atas 25 proses, yaitu: PO1, PO2, PO3, PO4, PO5, PO7, PO9, AI1, AI2, AI4, AI5, AI6, AI7, DS1, DS2, DS3, DS4, DS5, DS6, DS7, DS8, DS10, DS12, DS13, ME4.

Hasil pengukuran tingkat kematangan tata kelola TI diperoleh nilai kematangan 2,84 yaitu pada level *Define Process*, artinya sudah ada prosedur yang memiliki standar dan didokumentasikan dengan baik, sudah ada pelatihan formal untuk mengkomunikasikan prosedur dan kebijakan yang dibuat. Tetapi pada tahap implementasinya masih tergantung pada individu apakah bersedia melakukan prosedur yang ditetapkan atau tidak. Prosedur yang dibuat masih terbatas pada bentuk formalisasi dari praktik yang ada.

5. Daftar Pustaka

- [1] Riyanto Sarno, *Strategi Sukses Bisnis dengan Teknologi Informasi Berbasis Balanced Scorecard & COBIT*. Surabaya: ITS Press, 2009.
- [2] Jorge Robeiro dan Rui Gomes, "IT governance using COBIT implemented in a high public educational institution: a case study," In *Proceedings of the 3rd international conference on European computing conference*, hal. 41-52, 2009.
- [3] Dimas Riadi, "Pengukuran Tingkat Kematangan Proses Tata Kelola Teknologi Informasi Dengan Menggunakan Cobit 4.1 Maturity Model: Studi Kasus Dinas Pendidikan DKI Jakarta," Tesis 2013.
- [4] Andrea Pederiva, "The COBIT Maturity Model in a Vendor Evaluation Case," *Information Systems Control Journal USA*, vol. 3, pp. 2-3, 2003.
- [5] H.M.Jogiyanto, *Model Kesuksesan Teknologi Informasi*. Yogyakarta: C.V Andi Offset, 2007.
- [6] ISACA, *Integrating COBIT into the IT Audit Process (Planning, Scope Development, Practise)*: IT Governance Institute, 2006.
- [7] ISACA, *COBIT 4.1: Framework, Control Objective, Management Guidelines, Maturity Models*: IT Governance Institute, 2007.
- [8] S. Gondodiyoto, *Audit Sistem Informasi: Pendekatan Cobit*, Edisi Revisi ed. Jakarta: Mitra Wacana Media, 2007.
- [9] Christian J. Blunt dan Michael J. Hine, "Using COBIT to guide the adoption of Enterprise 2.0 technologies." *Journal of Applied Computing and Information Technology* vol. 7, no.1, 2009.
- [10] Zainal A.Hasibuan Ph.D, *Metodologi penelitian pada bidang ilmu komputer dan teknologi informasi*. Fakultas Ilmu Komputer Universitas Indonesia, 2007.

PERANCANGAN DAN PENGUJIAN *LOAD BALANCING* DAN *FAILOVER* MENGGUNAKAN NGINX

Rahmad Dani¹, Fajar Suryawan²

¹Program Studi Informatika

Universitas Muhammadiyah Surakarta

²Program Studi Teknik Elektro

Universitas Muhammadiyah Surakarta

*Fajar.Suryawan@ums.ac.id

Abstrak

Situs web dengan *traffic* yang tinggi dapat menyebabkan beban kerja yang berat di sisi server, yang pada gilirannya akan mengakibatkan turunnya kinerja server, bahkan kegagalan sistem secara keseluruhan. Salah satu solusi untuk mengatasi masalah tersebut adalah dengan menerapkan teknik *load balancing* dan *failover*. *Load balancing* merupakan teknologi untuk melakukan pembagian beban kepada beberapa server, memastikan tidak terjadi kelebihan beban pada salah satu server. Sementara itu, *failover* merupakan kemampuan suatu sistem untuk berpindah ke sistem cadangan jika sistem utama mengalami kegagalan. Dalam penelitian ini *load balancing* dengan teknik *failover* akan diimplementasikan pada sistem operasi Ubuntu. *Software* inti yang digunakan dalam penelitian ini adalah Nginx dan KeepAlived. Nginx akan berfungsi sebagai *load balancer*, sedangkan KeepAlived untuk mengimplementasikan teknik *failover*. Beberapa skenario telah disiapkan untuk menguji sistem *load balancing* yang telah dirancang. Pengujian dilakukan dengan menggunakan perangkat lunak JMeter. Berdasarkan pengujian yang telah dilakukan, sistem yang dirancang berhasil membagikan beban permintaan dan dapat terus bekerja walaupun terjadi kegagalan pada server *load balancer* ataupun kegagalan pada server *backend*. Selain itu, dalam beberapa pengujian, penggunaan *load balancing* terbukti mampu menurunkan waktu respons dan meningkatkan *throughput* pada sistem sehingga mampu meningkatkan performa keseluruhan sistem. Mengacu pada hasil penelitian ini, sistem *load balancing* dan *failover* menggunakan Nginx dapat dijadikan salah satu solusi pada sistem web server dengan situs web yang memiliki *traffic* tinggi.

Kata Kunci: load balancing, jaringan, failover, nginx, keepalived.

1. PENDAHULUAN

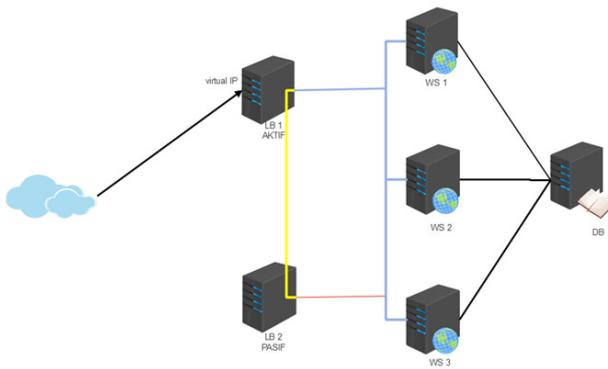
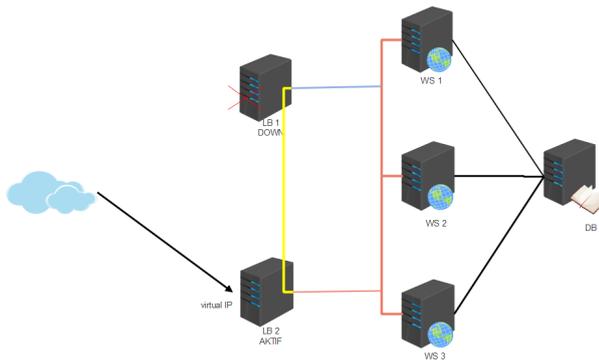
Seiring dengan perkembangan teknologi, jumlah penggunaan layanan web semakin meningkat. Hal ini menyebabkan situs-situs web populer memiliki jumlah *traffic* yang tinggi. Kegiatan atau acara tertentu juga dapat menyebabkan naiknya jumlah *traffic* web suatu organisasi. Contohnya, pada masa pendaftaran masuk kuliah atau sekolah *online*, pada masa KRS *online*, pada saat pemilu, atau pun *event-event* olahraga seperti piala dunia, olimpiade, dan *event-event* lain dapat menyebabkan peningkatan *traffic* web yang signifikan pada rentang waktu tersebut. Peningkatan jumlah *traffic* menyebabkan kerja server yang melayani permintaan menjadi semakin berat. Akibatnya performa server menurun dan sering terjadi gangguan pada layanan-layanan web tersebut. Jika tidak ditangani, hal ini dapat mengakibatkan sistem ataupun situs web tersebut mati/*down*.

Salah satu solusi untuk menangani masalah tersebut adalah dengan menggunakan sebuah server yang andal dengan performa yang tinggi. Solusi lain yang dapat diterapkan adalah teknologi *load balancing*. Teknologi *load*

balancing dapat membagi beban permintaan ke beberapa web server sehingga meringankan kinerja masing-masing server. Menurut Pandey, dkk (2015) *load balancing* merupakan aspek penting yang dapat mendistribusikan beban kerja ke banyak server secara optimal yang menghasilkan waktu tanggap yang baik dan meningkatkan tingkat kepuasan pengguna, meningkatkan efisiensi penggunaan sumber daya, dan berdampak pada peningkatan performa secara keseluruhan.

Sedangkan menurut Friedrich, Gairing, dan Sauerwald (2012), *load balancing* diperlukan untuk efisiensi kegunaan sumber daya komputer dalam sistem paralel dan terdistribusi. Tujuannya untuk merelokasi beban sehingga nantinya tiap *node* menerima jumlah beban yang sama.

Beberapa penelitian yang telah dilakukan menunjukkan bahwa *load balancing* mampu meningkatkan performa server. Handoko Yoga Hartomo (2015) mengatakan bahwa teknik *load balance* bekerja dengan cara membagi beban yang diterima oleh server dan ketika salah satu mati, maka server lain akan melayani permintaan dari pengguna. Penelitiannya juga turut membuktikan hasil penggunaan dua *software load balancing* pound dan haproxy di mana keduanya sama-sama mampu meningkatkan kemampuan server namun terbukti haproxy sedikit lebih unggul.

Gambar 1. Sistem *load balancing* utamaGambar 2. Sistem *load balancing* saat failover

Selain itu, Bayu Saputra (2012) juga menerapkan perancangan dan implementasi *load balancing* pada web server menggunakan haproxy, pengujian sistem dilakukan dengan membandingkan kinerja antara satu web server dengan sistem *load balancing* yang menggunakan dua web server. Hasilnya, beban kerja yang ditanggung pada sistem *load balancing* dibagi pada kedua server yang aktif sehingga kinerja server semakin cepat dan layanan tetap tersedia walaupun salah satu server *down*.

Dapat dikatakan bahwa penggunaan *load balancing* dapat mengatasi kegagalan yang terjadi pada sisi web server. Akan tetapi, jika *load balancing* mengalami kegagalan, seluruh layanan dapat terhenti. Oleh karena itu, diperlukan *failover* pada *load balancing* yang terpasang. Noviyanto, dkk (2015) menggunakan haproxy pada aplikasi web mendefinisikan bahwa *failover* merupakan kemampuan sebuah sistem untuk berpindah secara manual atau otomatis jika salah satu sistem mengalami kegagalan sehingga sistem lain menjadi *backup* bagi sistem yang mengalami kegagalan. Dengan cara ini sistem dapat terus berjalan walaupun sistem utama mengalami kegagalan.

Penelitian ini bertujuan untuk mengimplementasikan *load balancing* pada web server agar dapat meningkatkan kinerja sistem, serta mampu mengantisipasi kegagalan sistem melalui teknik *failover*. Manfaat dari penelitian ini adalah memahami konsep *load balancing* yang dipadu dengan teknik *failover* dengan menggunakan Nginx dan KeepAlived pada server berbasis Ubuntu.

2. METODE

Penelitian ini dilakukan secara eksperimental di Laboratorium Jaringan Komputer Fakultas Komunikasi dan Informatika Universitas Muhammadiyah Surakarta. Rancangan sistem *load balancing* dalam penelitian ini

menggunakan 6 buah komputer yang berperan sebagai server. Dari 6 komputer tersebut, 2 di antaranya berperan sebagai *load balancer*, 3 berperan sebagai web server, dan 1 berperan sebagai server *database*. Gambaran umum dari rancangan sistem *load balancing* dapat dilihat pada Gambar 1.

Gambar 1 menunjukkan sistem *load balancing* yang menggunakan metode aktif-pasif. Dalam metode aktif-pasif ini, hanya satu *load balancer* aktif/master yang akan melayani permintaan yang masuk. *Load balancer* lain atau *load balancer* pasif/slave hanya ikut memonitor kondisi *frontend* dan *backend* server. Apabila terjadi kegagalan pada *load balancer* yang aktif, sistem akan melakukan *failover* sehingga *load balancer* pasif akan mengambil alih status aktif dari master sementara waktu hingga master berfungsi kembali. Gambar 2 menunjukkan apa yang terjadi setelah *failover*.

Apabila sistem telah diimplementasikan, maka sistem siap diuji dalam beberapa skenario yang telah dirancang.

2.1 ALAT DAN BAHAN

Tahapan ini menentukan hal-hal yang dibutuhkan dalam proses penelitian seperti *hardware* dan *software*. Terdapat 7 komputer yang digunakan dalam penelitian ini, 6 sebagai server, dan 1 sebagai klien. Keenam komputer yang digunakan sebagai server tersebut memiliki spesifikasi *hardware* komputer sebagai berikut: Intel Core i3-2120 3.30GHz, 4GB DDR3, 500GB HDD dengan sistem operasi Ubuntu 12.10. Sedangkan komputer klien memiliki spesifikasi *hardware* sebagai berikut: Intel Core i3-2348 2.30GHz, 6GB DDR3, 500GB HDD dengan sistem operasi Windows 10. Untuk *software* yang digunakan, antara lain nginx, keepalived, jmeter, apache2, mysql, htop, saidar, nano, dan putty.

2.2 IMPLEMENTASI SISTEM

a. Pemberian IP Address

Proses konfigurasi dimulai dengan pengaturan ip *address* tiap-tiap komputer. Pada *load balancer*, IP *address* yang digunakan adalah 192.168.52.9/10. Sedangkan pada web server akan diberikan IP 192.168.52.2/6/12. Untuk server *database*, akan diberikan IP 192.168.52.16. Karena terdapat 2 *load balancer*, diberikan 1 virtual IP yang dapat berpindah antar *load balancer*. Virtual IP ini berfungsi sebagai IP dari situs web/aplikasi web yang digunakan. IP *address* untuk virtual IP adalah 192.168.52.99. Pengaturan IP *address* pada Ubuntu dapat dilakukan dengan cara mengubah isi file `$/etc/network/interface`. Hasil perubahan dapat dilihat dengan menggunakan perintah `$ifconfig`.

b. Konfigurasi Load Balancing Server

Pada kedua server *load balancing*, perlu dilakukan instalasi Nginx. Perintah instalasi Nginx melalui *repository* Ubuntu adalah `$apt-get install nginx`. Lalu dilakukan penambahan file konfigurasi baru dengan perintah `$sudo nano /etc/nginx/conf.d/lb.conf`. Nginx lalu di-*restart* agar perubahan diterapkan. File konfigurasi tersebut memiliki isi seperti ditunjukkan pada Gambar 3.

```

upstream myapp1 {
    #least_conn;
    server 192.168.52.2:80 max_fails=3 fail_timeout=30s;
    server 192.168.52.6:80 max_fails=3 fail_timeout=30s;
    server 192.168.52.12:80 max_fails=3 fail_timeout=30s;
}

server {
    listen 192.168.52.99:80;
    access_log /var/log/nginx/lb.access.log;
    error_log /var/log/nginx/lb-error.log error;

    location / {
        proxy_pass http://myapp1;
    }
}

```

Gambar 3. Konfigurasi lb.conf

```

#
# Log Martian Packets
#net.ipv4.conf.all.log_martians = 1
#
net.ipv4.ip_nonlocal_bind=1

```

Gambar 4. Penambahan slot ip.

```

LogFormat "%v:%p %s(X-Forwarded-For)i %l %u %t \"%r\" %>s %O \"%s(Referer)i\" \"%s(User-Agent)i\" " vhost_combined
LogFormat "%s(X-Forwarded-For)i %l %u %t \"%r\" %>s %O \"%s(Referer)i\" \"%s(User-Agent)i\" " combined
LogFormat "%s(X-Forwarded-For)i %l %u %t \"%r\" %>s %O " common
LogFormat "%s(Referer)i -> %U" referer
LogFormat "%s(User-agent)i" agent

```

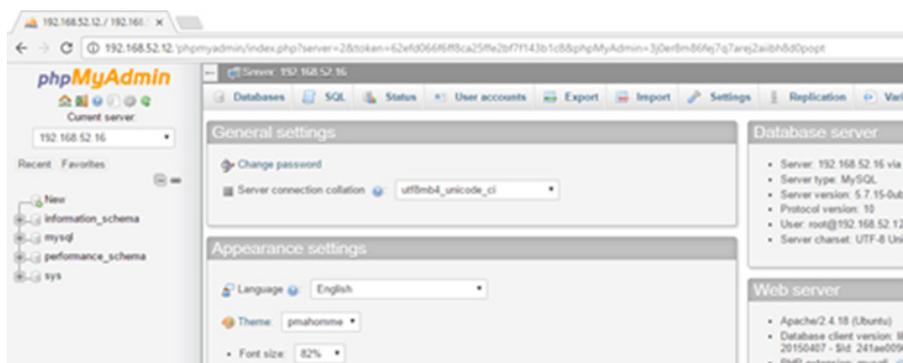
Gambar 5. Konfigurasi keepalived.conf

```

GNU nano 2.2.6 File: /etc/keepalived/keepalived.conf
vrrp_script virtual_ip {
    script "pidof nginx"
    interval 2
}
vrrp_instance VI_1 {
    interface eth0
    state MASTER
    virtual_router_id 51
    priority 101
    virtual_ipaddress {
        192.168.52.99
    }
    track_script {
        virtual_ip
    }
}

```

Gambar 6. Konfigurasi file apache2.conf



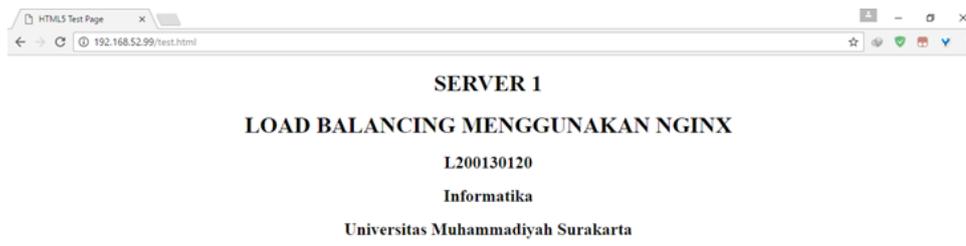
Gambar 7. Akses server database menggunakan phpmyadmin.

c. Konfigurasi Failover

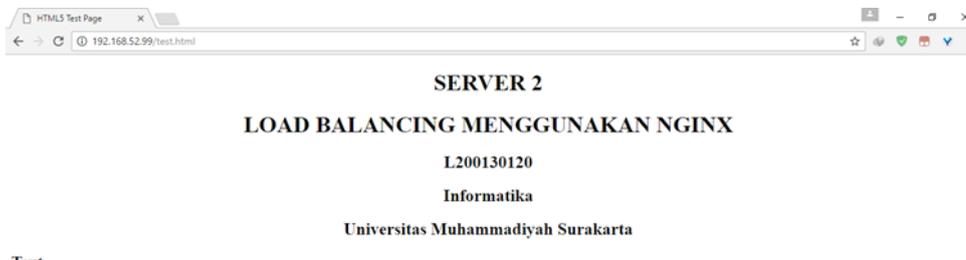
Konfigurasi *failover* dimulai dengan menambahkan slot untuk ip virtual pada kedua server *load balancing*. Caranya dengan menambahkan baris `net.ipv4.ip_nonlocal_bind=1` pada file `$/etc/sysctl.conf`. Agar perubahan diterapkan, masukkan perintah `$sudo sysctl -p`.

Selanjutnya, *keepalived* diinstal dengan perintah `$sudo apt-get install keepalived`. Digunakan perintah

`$sudo nano /etc/keepalived/keepalived.conf` untuk membuat file konfigurasi baru. Konfigurasi tersebut disimpan, lalu *keepalived* di-restart. Terdapat perbedaan pada kedua file konfigurasi *KeepAlived* di kedua *load balancer*. Perbedaan tersebut adalah pada angka *priority*. *Priority* pada server *load balancing* master harus lebih tinggi dibandingkan pada server *load balancing slave*, tujuannya adalah agar kedua *load balancing* tidak memiliki ip virtual secara bersamaan.

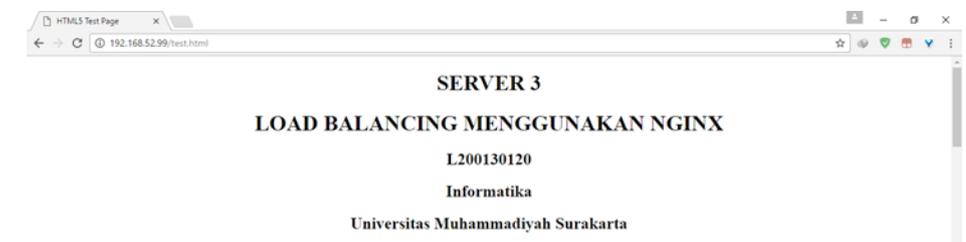


Gambar 8. Server 1 melayani pengguna



Text

Gambar 9. Server 2 melayani pengguna



Gambar 10. Server 3 melayani pengguna

Tabel 1. Pembagian beban

No	Total Request	Server 1	Server 2	Server 3
1	10	3	4	3
2	50	17	16	17

- d. Konfigurasi Server Backend dan Server Database
Server *backend* dalam penelitian ini menggunakan apache2, php5, dan phpmyadmin. Agar *backend* dapat melihat ip klien, telah diedit kata `%h` menjadi `%{X-Forwarded-For}` pada file konfigurasi `$/etc/apache2/apache2.conf`.

Pada server *database software* yang digunakan adalah mysql. Selain mudah, mysql dapat diakses secara remote melalui phpmyadmin.

2.3 PENGUJIAN SISTEM

Pada tahapan ini, sistem diuji menggunakan aplikasi JMeter untuk melakukan simulasi pemberian beban dari komputer yang bertindak sebagai klien. Saat melakukan pengujian, peneliti memonitoring keadaan server-server terutama server *backend* untuk mengetahui performa dari server *backend* dengan bantuan beberapa *software monitoring* seperti `saidar`, `htop`, `top`, dan juga beberapa perintah dasar pada linux seperti `tail`, `awk`, `grep`. Hasil-hasil dari uji coba tersebut akan dimuat dalam laporan beserta analisis dari peneliti.

Dalam pengujian, hasil pemasangan *load balancing system* ini dapat diakses melalui *browser* dengan alamat ip virtual yang terpasang yaitu `http://192.168.52.99/test`.

html. Untuk melihat server mana yang sedang melayani *user*, maka pada sisi *backend* dapat dipasang konten berbeda.

3. HASIL PENGUJIAN, PEMBAHASAN, DAN DISKUSI

3.1 PENGUJIAN PEMBAGIAN BEBAN

Pengujian ini dilakukan dengan cara mensimulasikan sejumlah klien yang mengakses sistem *load balancing* di mana setiap satu klien melakukan sekali *request*. Semua *request* tersebut, direkam oleh *log apache* pada *backend-backend* server. Data-data pada *log apache* tersebut dihitung dengan menggunakan perintah `grep`. Hasilnya dapat dilihat pada tabel berikut.

Sebelum setiap pengujian, isi *log apache* pada setiap *backend* server dihapus. Berdasarkan tabel 1, pada pengujian pertama menggunakan 10 *request*, urutan pelayanan *request* dimulai dari server 2, server 3, server 1 dan terus berulang sampai semua *request* terlayani. Hasilnya menunjukkan bahwa server 1 melayani 3 *request*, server 2 melayani 4 *request*, sedangkan server 3 melayani 3 *request*. Pada uji coba kedua dengan 50 *request*, urutan pelayanan *request* adalah server 3, server 1, server 2. Dimana server 1 melayani 17 *request*, server 2 melayani 16 *request* dan server 3 melayani 17 *request*. Dari hasil tersebut terlihat bahwa *load balancer* membagikan beban secara merata secara berurutan sesuai dengan algoritma *round robin*.

Tabel 5. Hasil uji dengan laman php

No	User	Total request	Sistem yang digunakan	Waktu respons rata-rata (ms)	Throughput (request/s)	Error (%)
1	400	10000	Tunggal	356	256,8	0
			LB	14	327,9	0
2	800	20000	Tunggal	1525	261,0	1,01
			LB	17	651,5	0
3	1200	30000	Tunggal	2840	218,1	4,80
			LB	349	760,0	0

Tabel 2. Ketersediaan saat terjadi kegagalan pada sisi *backend*

No Uji	Total Request	Server 1	Server 2	Server 3
1	10	5	5	down
2	10	5	down	5
3	10	Down	down	10

Tabel 3. Hasil pengujian *failover* pada *load balancer*

No	Request	Berhasil	Error	Keterangan
1	100	98	2	LB 1 ke LB 2
2	100	100	0	LB 2 ke LB 1

Tabel 4. Hasil uji dengan laman html

No	User	Total Request	Sistem yang Digunakan	Waktu Respons Rata-Rata (ms)	Throughput (request/s)	Error
1	400	10000	Tunggal	3	330,5	0
			LB	4	331,6	0
2	800	20000	Tunggal	3	657,1	0
			LB	7	658,3	0
3	1200	30000	Tunggal	93	907,0	0
			LB	239	805,7	0

3.2 PENGUJIAN KETERSEDIAAN LAYANAN

Pengujian selanjutnya yaitu dengan cara mematikan salah satu *backend* server yang ada. Pengujian ini dilakukan dengan memberikan 10 *request* kepada sistem yang sebelum pengujian dilakukan, satu atau dua *backend* server dimatikan. Tujuannya adalah untuk melihat apakah akan terjadi gangguan pada pelayanan oleh *backend* server.

Berdasarkan Tabel 2, pada pengujian pertama, server 3 dalam keadaan mati sehingga semua *request* dilayani oleh server 1 dan server 2. Pada pengujian kedua, server 2 dalam keadaan mati sehingga semua *request* dilayani oleh server 1 dan server 3. Pada pengujian ketiga, server 1 dan server 2 dalam keadaan mati sehingga semua *request* dilayani oleh server 3. Terbukti bahwa **apabila salah satu *backend* server mengalami kegagalan, hal ini tidak akan mengganggu sistem yang sedang berjalan sehingga pengguna tetap mendapatkan pelayanan dari sistem.**

Pengujian selanjutnya dilakukan untuk menguji *failover* pada *load balancing* sistem. Pengujian ini dilakukan dengan cara memberikan beban sebanyak 100 *request* dengan ketentuan 1 *request* perdetik selama 100 detik di mana pada pertengahan uji, server *load balancer* yang aktif akan sengaja dimatikan dan memaksa terjadinya *failover* ke server *load balancer* pasif. Berikut hasil percobaannya.

Uji coba pertama dari Tabel 3 dilakukan dengan cara mematikan server *load balancer* 1 saat sedang dilakukan pemberian *request*. Hasilnya menunjukkan bahwa ketika sedang terjadi perpindahan sistem dari server *load balancer* 1 ke server *load balancer* 2, *request* tidak dapat masuk karena sistem sedang *offline*. Pada percobaan pertama, sistem memerlukan waktu 5 detik untuk berpindah dari *load balancer* 1 ke *load balancer* 2 dan selama perpindahan tersebut terdapat 2 *request* oleh user yang tidak dilayani. Pada pengujian kedua, perpindahan sistem terjadi dari server *load balancer* 2 ke server *load balancer* 1. Pada pengujian ini, tidak terjadi *error* sama sekali yang menunjukkan bahwa perpindahan sistem dari sistem cadangan kembali ke sistem utama tidak akan mempengaruhi jalannya sistem web. Pengujian ini membuktikan bahwa **sistem *load balancing* yang terpasang mampu melakukan *failover* untuk mengantisipasi kegagalan pada server *load balancer*.**

3.3 PENGUJIAN KINERJA SERVER

Pengujian Kinerja Server merupakan pengujian untuk mengamati performa *backend* server dalam keadaan diberi beban yang cukup banyak. Tujuan dari pengujian ini adalah untuk melihat perbandingan performa setelah menggunakan *load balancing* dengan sebelum menggunakan

Tabel 6. Hasil uji input data dengan php

No	User	Total request	Sistem yang digunakan	Waktu respon rata-rata (ms)	Throughput (request/s)	Error (%)
1	400	10000	Tunggal	185	288,6	0
			LB	172	285,6	0
2	800	20000	Tunggal	247	571,6	0
			LB	187	576,7	0
3	1200	30000	Tunggal	477	658,6	0.32
			LB	196	836,9	0

pada *load balancing*. Performa dihitung berdasarkan waktu respons yaitu waktu yang diperlukan sistem untuk melayani *request* dan juga *throughput* yaitu banyaknya *request* yang dapat dilayani sistem tiap detiknya. JMeter akan merekam kedua hasil tersebut.

3.4 PENGUJIAN DENGAN LAMAN HTML MURNI

Pada pengujian ini, dilakukan dengan cara memberikan sejumlah *request* pada sistem web yang didalamnya telah dipasang laman web html seberat 12 kB. Pengujian ini dilakukan dengan cara membandingkan hasil saat menggunakan *load balancing* dan hasil saat menggunakan server web tunggal. Berikut adalah hasil dari pengujian ini.

Berdasarkan Tabel 4, pengujian pertama menggunakan 400 *concurrent* user dengan total 10000 *request* yang diberikan dalam kurun waktu 30 detik. Terlihat bahwa rata-rata *response time* saat menggunakan server tunggal adalah 3 ms untuk melayani setiap *request* masuk. Hasil ini lebih baik jika dibanding dengan penggunaan *load balancing* yang mencapai 4 ms untuk tiap *request*-nya. Sedangkan *throughput* tidak banyak perbedaan di antara keduanya yaitu berkisar pada angka 331 *request* per detik. Pada pengujian kedua, dengan menggunakan 800 *concurrent* user dengan *Total request* sebanyak 20.000 yang diberikan dalam kurun waktu 30 detik, terlihat bahwa waktu rata-rata *request* dilayani oleh server tunggal adalah 3 ms sedangkan untuk *load balancing* membutuhkan waktu 7 ms. Untuk *throughput* keduanya memiliki jumlah yang relatif sama yaitu sekitar 657 *request* per detik. Pada pengujian ketiga dengan 1.200 *concurrent* user dan *total request* sebanyak 30.000 yang diberikan dalam kurun waktu 30 detik, menghasilkan waktu respons sebesar 93 ms pada server tunggal. Pada saat menggunakan *load balancing*, waktu respons yang didapatkan tidak lebih baik yaitu sebesar 239 ms. Untuk *throughput* pada server tunggal juga lebih baik yaitu sebesar 907 *request* per detik jika dibandingkan dengan *load balancing* yang sebesar 805,7 *request* per detik. Dalam pengujian ini, penggunaan *load balancing* pada sistem terbukti tidak memberikan dampak lebih baik.

3.5 PENGUJIAN DENGAN LAMAN PHP

Pengujian ini dilakukan dengan memberikan sejumlah *request* kepada sistem web yang di dalamnya telah terpasang sebuah *file* php. *File* php tersebut akan melakukan perhitungan untuk mencari nilai pi (π) di sisi server dan hasilnya akan dikirim ke user sebagai respons dari permintaan yang dilakukan *user*. Pengujian ini membandingkan antara hasil dengan menggunakan sistem *load balancing* dan hasil menggunakan server web tunggal. Berikut ini merupakan hasil dari pengujian-pengujian tersebut.

Berdasarkan Tabel 5, pengujian pertama menggunakan 400 *concurrent* user dengan total 10000 *request* yang dikirim dalam kurun waktu 30 detik. Waktu respons rata-rata sistem untuk melayani *request* yang masuk saat menggunakan server tunggal adalah sebesar 356 ms. Sedangkan waktu respons rata-rata saat menggunakan *load balancing* menunjukkan hasil yang lebih baik yaitu sebesar 14 ms. *Throughput* saat menggunakan server tunggal hanya mencapai angka 256,8 *request* per detik jika dibanding dengan saat menggunakan *load balancing* yang sebesar 327,9 *request* per detik. Pada pengujian kedua dengan menggunakan 800 *concurrent* user yang melakukan *total request* sebanyak 20.000 yang diberikan pada kurun waktu 30 detik, hasilnya menunjukkan waktu respons rata-rata pada server tunggal sebesar 1.525 ms. Sedangkan saat menggunakan *load balancing*, waktu respons rata-rata sebesar 17 ms. *Throughput* pada server tunggal menunjukkan angka 261 *request* per detik sedangkan saat menggunakan *load balancing* hasilnya adalah 651,5 *request* per detik. Pada server tunggal 1,01% *request* dari *Total request* mengalami *error* sedangkan saat menggunakan *load balancing* tidak terdapat *request* yang mengalami *error*. Pengujian ketiga dengan menggunakan 1.200 *concurrent* user dan *Total request* sebanyak 30.000 yang diberikan dalam kurun waktu 30 detik menunjukkan bahwa waktu respons rata-rata saat menggunakan server tunggal adalah 2.840 ms. Sedangkan waktu respons rata-rata saat menggunakan *load balancing* adalah 349 ms. *Throughput* pada server tunggal menunjukkan bahwa sistem mampu melayani 218,1 *request* per detik, sedangkan saat menggunakan *load balancing*, sistem mampu melayani 760 *request* per detik. Pengujian ketiga pada server tunggal menunjukkan bahwa terdapat 4,80% *request* dari *total request* yang tidak terpenuhi sedangkan saat menggunakan *load balancing*, tidak ada *request* yang gagal terpenuhi. Hasil pengujian-pengujian dengan laman php menunjukkan bahwa penggunaan sistem *load balancing* memiliki hasil lebih baik.

3.6 PENGUJIAN DENGAN LAMAN PHP DAN DATABASE

Pengujian ini dilakukan dengan cara memberikan sejumlah *request* pada sistem web untuk melakukan input data ke dalam *database* mysql melalui php. Dalam pengujian ini, data yang dimasukkan berupa 200 karakter *string*, *timestamp*, dan angka. Setiap *request* akan melakukan input data dua kali ke dalam *database*. Hasil dari pengujian ini adalah sebagai berikut.

Berdasarkan Tabel 6, hasil dari pengujian pertama dengan menggunakan 400 *concurrent* user dengan *total request* sebanyak 10.000 yang diberikan dalam waktu 30 detik menunjukkan bahwa waktu respons rata-rata menggunakan server tunggal adalah 185 ms sedangkan

Tabel 7. Hasil uji input gambar ke mysql

No	User	Total request	Waktu Respon (ms)	Throughput (request/s)	Error (%)
1	100	2000	12302	6,8	0
2	200	4000	22241	8	20,42
3	300	6000	3619	62,5	89,32

waktu respons rata-rata saat menggunakan *load balancing* adalah 172 ms. *Throughput* yang didapat ketika menggunakan server tunggal adalah 288,6 *request* per detik dan saat menggunakan *load balancing* memiliki hasil 285,6 *request* per detik. Pada pengujian kedua dengan menggunakan 800 *concurrent user* dan juga *total request* sebanyak 20.000 yang diberikan dalam kurun waktu 30 detik menunjukkan bahwa waktu respons rata-rata saat menggunakan server tunggal adalah sebesar 247 ms sedangkan saat menggunakan *load balancing* waktu respons rata-rata sebesar 187 ms. *Throughput* pada server tunggal mencapai angka 571,6 *request* per detik sedangkan *throughput* saat menggunakan *load balancing* hanya sebesar 576,7 *request* per detik. Pada pengujian ketiga, menggunakan 1200 *concurrent user* dengan total 30000 *request* yang diberikan dalam kurun waktu 30 detik menunjukkan bahwa waktu respon rata-rata saat menggunakan server tunggal adalah 477 ms sedangkan saat menggunakan *load balancing* adalah 196 ms. *Throughput* pada server tunggal menunjukkan angka sebesar 658,6 *requests*/detik sedangkan saat menggunakan *load balancing* hasil *throughput* menunjukkan angka 836,9 *request* per detik. Pada pengujian ketiga ini, saat menggunakan server tunggal terdapat 0.32% *request* dari *Total request* yang mengalami kegagalan, sedangkan saat menggunakan *load balancing*, tidak terdapat *request* yang mengalami kegagalan. Pada pengujian ini, **pemasangan *load balancing system* terbukti memberikan peningkatan kinerja.**

Pada pengujian selanjutnya, dilakukan dengan cara memasukkan gambar ke dalam *database* mysql. Gambar yang dimasukkan memiliki ekstensi jpg, tetapi pada mysql gambar akan memiliki format blob file. Ukuran gambar yang dimasukkan adalah sekitar 500 KB. Setiap *request* yang dilakukan oleh pengguna dalam pengujian kali ini akan memasukkan dua buah gambar. Perlu diketahui bahwa pengujian ini dilakukan setelah mysql dioptimasi dengan cara memperbesar jumlah koneksi yang mampu dilayani mysql tiap waktu dan juga memperbesar ukuran *log file* untuk innoDB. Jumlah koneksi tersebut diperbesar dari aturan *default* sebesar 150 menjadi 500, sedangkan ukuran *log file* diperbesar menjadi 128 MB. Hal ini dilakukan karena sebelum dioptimasi, pada sisi *backend* server sering mendapat peringatan “*to many connections*”, sedangkan pada sisi server *database* mendapat *error* dengan keterangan bahwa ukuran *log files* tidak cukup. Berikut merupakan hasil dari pengujian ini:

Berdasarkan hasil yang terlihat pada Tabel 7, pada pengujian pertama dengan 100 *concurrent user* dengan *total request* sebanyak 2000 *request* yang diberikan dalam waktu 30 detik, waktu respons rata-rata tiap *request* yang diberikan oleh *user* adalah sekitar 12.302 ms dengan *throughput* sebanyak 6,8 *request* per detiknya. Pada pengujian kedua, dengan 200 *concurrent user* yang memberikan *Total request* sebanyak 4000 dalam waktu 30 detik menghasilkan waktu respons sebesar 22.241 ms dengan *throughput* sebesar 8 *request* per detik. Dalam pengujian kedua, sekitar 20% *request* yang diberikan tidak/gagal dilayani oleh server.

Pada pengujian ketiga dengan 300 *concurrent user* dengan *total request* sebanyak 6.000 dalam 30 detik memberikan hasil berupa waktu respons sebesar 3.619 ms dengan *throughput* sebesar 62,5 *request* per detik. *Request* yang gagal dilayani pada percobaan ketiga mencapai 89% dari *total request* yang diberikan. Pengujian input gambar ini membuktikan bahwa mysql tidak mampu melayani query berat dari 3 server *backend* di depannya walaupun telah dilakukan optimasi.

4. PENUTUP

Berdasarkan hasil pengujian dan pembahasan yang telah dilakukan dalam penelitian ini, dapat ditarik beberapa kesimpulan yaitu:

- Berdasarkan pengujian, sistem yang dirancang dapat membagikan beban secara merata ke beberapa *backend* server baik dalam keadaan semua server normal atau pun saat terjadi kegagalan pada salah satu *backend* server.
- Sistem yang dirancang dapat meningkatkan ketersediaan karena mampu melakukan *failover* saat terjadi kegagalan baik di sisi *load balancer* atau pun di sisi *backend* server.
- Berdasarkan pengujian menggunakan jmeter yang telah dilakukan, nginx *load balancing* tidak meningkatkan waktu respons dan *throughput* pada *request* terhadap laman html. Akan tetapi saat melayani *request* laman php atau pun input data menggunakan php ke mysql, nginx *load balancing* mampu meningkatkan waktu respons dan *throughput* dengan baik.
- Berdasarkan penelitian ini, penggunaan *load balancing* membuat sisi *database* menjadi lebih berat karena hanya menggunakan satu buah *database* sehingga saat menjalankan *request* yang memerlukan query yang berat, banyak *request* yang gagal.
- Penggunaan *load balancing* dan *failover* nginx mampu menstabilkan dan menjaga performa sistem dapat dijadikan solusi untuk situs dengan *traffic* tinggi sehingga tidak terjadi kegagalan pada sistem dan aplikasi web yang terpasang pada sistem.

Saran bagi penelitian serupa yang akan datang adalah untuk dapat mengimplementasikan rancangan ke server dengan kondisi lingkungan yang nyata. Selain itu, penelitian selanjutnya disarankan untuk menggunakan lebih dari satu *database* untuk menopang sistem *load balancing* di depannya.

5. DAFTAR PUSTAKA

- [1] Saputra, Ilham Bayu. (2012). Perancangan dan Implementasi *Load Balancing* Web Server Menggunakan *HAProxy*. Skripsi. Surakarta: Universitas Muhammadiyah Surakarta.
- [2] Friedrich, T., Gairing, M., & Sauerwald, T.

- (2012). *Quasirandom load balancing*. *SLAM Journal on Computing*, 41(4), 747-771. doi:<http://dx.doi.org/10.1137/100799216>
- [3] Pandey, Shilipi. dkk (2015). *Load Balancing Techniques: A Comprehensive Study*. IJARCSMS: Volume 3 Issue 4.
- [4] Noviyanto, Ari Budi. dkk. (2015). Perancangan dan Implementasi *Load Balancing Reverse Proxy* Menggunakan HAProxy pada Aplikasi Web. Jurnal JARKOM, vol 3 no 1.
- [5] Hartomo, Handoko Yoga. (2015). Implementasi Web Server *Load balancing* pada Mesin Virtual. Skripsi. Surakarta: Universitas Muhammadiyah Surakarta.
- [6] Nginx Documentation (2016). *Using Nginx as Load Balancing* http://nginx.org/en/docs/http/load_balancing.html (diakses pada 25 November 2016)
- [7] Jmeter Documentation (2016). *Component Reference* http://jmeter.apache.org/usermanual/component_reference.html#Aggregate_Report (diakses pada 30 November 2016)

Mitra bestari (*Reviewers*)

Dewan editor mengucapkan terima kasih yang sebesar-besarnya atas partisipasi para *reviewer* berikut ini yang telah membantu dalam proses penerbitan jurnal Khazanah Informatika volume 3 nomor 1, Juni 2017.

1. Sri Karnila, Informatics and Business Institute Darmajaya, Bandar Lampung
2. Adi Supriyatna, AMIK BSI Karawang
3. Aris Rakhmadi, UMS
4. Azizah Fatmawati, UMS
5. Dedi Gunawan, UMS
6. Dr. Bana Handaga, UMS
7. Dr. Fajar Suryawan, UMS
8. Dr. Gunawan Ariyanto, UMS
9. Dr. Nurgiyatna, UMS
10. Endah Sudarmilah, UMS
11. Endang Wahyu Pamungkas, UMS
12. Fatah Yasin Al Irsyadi, UMS
13. Jan Wantoro, UMS
14. Sukirman, UMS
15. Umi Fadlilah, UMS
16. Yogie Indra Kurniawan, UMS
17. Yusuf Sulisty Nugroho, UMS