



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*)

Didiek Wiyono, Universitas Sebelas Maret
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
Asslia Johar Latipah, Universitas Muhammadiyah Kalimantan Timur
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 ijinnya sehingga jurnal Khazanah Informatika volume 4 nomor 1 bulan Juni 2018 dapat diterbitkan. Semoga penerbitan ini menambah koleksi dan direktori ilmu pengetahuan khususnya pada bidang ilmu komputer dan informatika.

Terbitan kali ini berisi sembilan artikel. Satu artikel ditulis oleh penulis dari Universitas Muhammadiyah Surakarta, dan tujuh artikel lainnya ditulis oleh penulis dari berbagai perguruan tinggi di Indonesia, termasuk di antaranya artikel dari perguruan tinggi di Sumatera dan Kalimantan.

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 2018, kami mendapat kiriman 24 artikel, 8 artikel diputuskan untuk diterbitkan, 3 masih dalam proses review (untuk diterbitkan pada edisi mendatang jika diterima) dan 12 artikel tidak diteruskan ke proses penerbitan. Lama proses review sebuah artikel adalah rata-rata 32 hari sejak dikirim hingga dinyatakan diterima.

Mutu jurnal ini telah diakui oleh **lembaga akreditasi jurnal nasional (Arjuna) yang menempatkan Khazanah Informatika pada peringkat Sinta S2**. Peringkat S2 pada masa lalu diidentikkan dengan jurnal terakreditasi B. Selain itu reputasi jurnal Khazanah Informatika telah diakui lembaga internasional karena sejak Oktober 2016, jurnal telah terindeks oleh DOAJ (Directory of Open Access Journal).

Salam Hangat

Chief Editor

Daftar Isi

Penerapan Algoritma C4.5 pada Aplikasi Prediksi Kelulusan Mahasiswa Prodi Informatika	
<i>Ratna Puspita Sari Putri, Indra Waspada</i>	1-7
Perancangan Aplikasi <i>Chatting</i> Berbasis <i>Web</i> di PT. Pura Barutama Kudus Menggunakan Socket.IO dan <i>Framework Foundation</i>	
<i>Ramos Somya</i>	8-15
Prototype Sistem Informasi Pelayanan Bayi Baru Lahir pada Fasilitas Kesehatan Primer	
<i>Rinda Nurul Karimah, Andri Permana Wicaksono</i>	16-20
Rekomendasi Paket Produk Guna Meningkatkan Penjualan Dengan Metode FP-Growth	
<i>Asrul Abdullah</i>	21-26
Sistem Pendukung Keputusan dengan Metode AHP untuk Penilaian Kompetensi <i>Soft Skill</i> Karyawan	
<i>Rusydi Umar, Abdul Fadlil, Yuminah</i>	27-34
Peningkatan Kinerja Jaringan Komputer dengan <i>Border Gateway Protocol</i> (BGP) dan <i>Dynamic Routing</i> (Studi Kasus PT. Estiko Ramanda)	
<i>Tati Ernawati1, Jemi Endrawan</i>	35-41
Pelacak Orang Hilang Menggunakan Sepatu dengan Sistem GPS dan GSM	
<i>Mayang Rizqi Ambagapuri, Feby Nurkhalih Sulistya Putra, Mifta Thabira, Umi Fadlilah</i>	42-46
Aplikasi Sistem Pakar Pendeteksi Penyakit Tuberkulosis Berbasis Android	
<i>Eko Didik Widianto, Yuni Waz Zaituun, Ike Pertivi Windasari</i>	47-54

Penerapan Algoritma C4.5 pada Aplikasi Prediksi Kelulusan Mahasiswa Prodi Informatika

Ratna Puspita Sari Putri*, Indra Waspada

Departemen Ilmu Komputer/Informatika, Fakultas Sains dan Matematika

Universitas Diponegoro

Semarang

*ratnaps@gmail.com

Abstrak-Data tentang mahasiswa yang lulus merupakan sebuah data yang penting baik bagi departemen, fakultas maupun universitas karena data tersebut digunakan dalam proses akreditasi. Data tentang mahasiswa yang lulus terus bertambah di tiap tahunnya dan menumpuk seperti data yang terabaikan karena jarang digunakan. Data tentang mahasiswa yang lulus dapat memberikan informasi yang berguna jika dimanfaatkan dengan maksimal. Maka dari itu, penelitian ini akan memanfaatkan data tentang mahasiswa yang lulus dengan mengolahnya menggunakan *data mining* untuk mendapatkan informasi berupa prediksi kelulusan mahasiswa. Metode yang akan digunakan adalah metode pohon keputusan yang dibangun dengan algoritma C4.5 disertai dengan algoritma *error-based pruning* untuk proses pemotongan pohon keputusan. Kriteria yang akan digunakan adalah jenis kelamin, asal daerah, IPK, dan TOEFL. Dalam penerapannya, algoritma C4.5 dapat digunakan untuk menghasilkan prediksi kelulusan dengan nilai rata-rata *precision* 63.93%, *recall* 60.73%, dan akurasi **60.52%**. Setelah pohon keputusan dipotong dengan menggunakan metode *error-based pruning*, didapatkan hasil yang lebih baik. Pohon yang dipotong dengan menggunakan nilai *confidence* 0,4 menghasilkan *precision* 70.70%, *recall* 50.65%, dan akurasi 61.57%. Sedangkan pohon yang dipotong dengan menggunakan nilai *confidence* 0,25 menghasilkan *precision* 73.77%, *recall* 48.84%, dan akurasi 62.44%.

Kata Kunci: *data mining*, kelulusan mahasiswa, pohon keputusan, C4.5, *error-based pruning*

1. Pendahuluan

Di era digital ini banyak instansi dan perusahaan yang telah menyimpan data mereka di dalam sebuah *database* yang terkomputerisasi. Dunia pendidikan pun tidak terlepas dari perkembangan teknologi ini. Universitas Diponegoro termasuk salah satu perguruan tinggi yang telah menyimpan datanya dalam *database* yang terkomputerisasi. Data tersebut merupakan data mahasiswa, data dosen, serta berbagai data lain yang berhubungan dengan Universitas Diponegoro.

Data tersebut tidak banyak memiliki kegunaan dan seolah-olah menjadi sekumpulan data terabaikan yang bertambah besar tiap tahunnya. Data tersebut hanya digunakan saat universitas membutuhkan suatu informasi tertentu atau saat proses akreditasi. Saat mahasiswa telah lulus maka data mereka akan semakin jarang digunakan. Padahal data tentang mahasiswa yang lulus merupakan data yang penting dan digunakan dalam proses akreditasi.

Data tentang mahasiswa yang lulus dapat memberikan informasi yang berguna bagi universitas jika dimanfaatkan dengan maksimal. Salah satu cara untuk memanfaatkan data tentang mahasiswa yang lulus ini adalah dengan mengolahnya menggunakan *data mining*. Dengan proses *data mining* ini dapat ditemukan pola atau aturan yang dapat digunakan untuk menghasilkan suatu informasi seperti prediksi kelulusan mahasiswa.

Prediksi kelulusan mahasiswa dapat digunakan lebih lanjut untuk membantu universitas dalam mengevaluasi dan memperbaiki sistem pembelajaran sehingga universitas dapat menghasilkan lulusan yang berkualitas.

Penelitian ini akan dilaksanakan di Prodi Informatika. Prodi Informatika telah berdiri dari tahun 2004 dan memiliki sasaran untuk menjadi program studi unggulan. Oleh karena itu hasil prediksi kelulusan mahasiswa dapat membantu Prodi Informatika dalam mengambil langkah strategis.

Dalam penelitian yang berjudul *Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques*, Jadhav dan Channe membandingkan performa metode K-NN, Naive Bayes, dan pohon keputusan dalam berbagai aspek dengan menggunakan berbagai *dataset*. Dari hasil penelitian tersebut dapat diketahui bahwa pohon keputusan merupakan metode yang paling cepat performanya dibandingkan dengan metode yang lain. Selain itu pohon keputusan lebih akurat dan memiliki *error rate* yang rendah [1].

Dalam penelitian lain yang berjudul *Comparative Analysis of Decision Tree Algorithms for The Prediction of Eligibility of A Man for Availing Bank Loan*, Mohankumar dkk membandingkan berbagai algoritma untuk membangun pohon keputusan dan algoritma C4.5 merupakan algoritma dengan performa tercepat dan memiliki akurasi yang paling tinggi [2].

David Kamagi mengimplementasikan algoritma C4.5 dalam penelitian berjudul "Implementasi Data Mining dengan Algoritma C4.5 untuk Memprediksi Tingkat Kelulusan Mahasiswa" dan menghasilkan prediksi dengan tingkat keakuratan yang tinggi, yaitu 87,5% [3]. Kamagi menggunakan empat kelas target, yaitu lulus cepat, lulus tepat, lulus terlambat, dan *drop out*. Atribut yang digunakan adalah IPS, jenis kelamin, asal sekolah, tipe kelulusan, dan

jumlah SKS. Dalam penelitian ini akan digunakan kelas target <5 tahun dan ≥ 5 tahun. Sedangkan atribut yang digunakan adalah IPK, TOEFL, asal daerah, dan jenis kelamin.

Berdasarkan penjelasan di atas akan dibangun sebuah aplikasi prediksi kelulusan mahasiswa Prodi Informatika dengan menggunakan metode pohon keputusan yang dibangun menggunakan algoritma C4.5.

2. Metode

a. Pemahaman Domain dan Tujuan KDD

Data tentang mahasiswa yang lulus merupakan sebuah data yang penting karena data tersebut digunakan dalam proses akreditasi. Dalam penelitian sebelumnya telah dibangun sebuah aplikasi repositori lulusan yang digunakan untuk menyimpan dan mengelola data tentang mahasiswa Prodi Informatika yang lulus. Setelah disimpan data tersebut tidak terlalu banyak digunakan dan semakin bertambah besar tiap tahunnya.

Data tentang mahasiswa yang lulus dapat memberikan informasi yang berguna bagi prodi jika dimanfaatkan dengan maksimal. Salah satu cara untuk memanfaatkan data tentang mahasiswa yang lulus ini adalah dengan mengolahnya menggunakan *data mining*. Dengan proses *data mining* ini dapat ditemukan pola atau aturan yang dapat digunakan untuk menghasilkan suatu informasi seperti prediksi tepat atau tidaknya kelulusan mahasiswa.

Untuk itu akan dilakukan proses penggalian informasi dari data tentang mahasiswa yang lulus Prodi Informatika dengan menggunakan model *knowledge discovery in databases* (KDD). Tujuan yang diharapkan dari proses KDD ini adalah mendapatkan informasi mengenai kelulusan seorang mahasiswa berdasarkan jenis kelamin, asal daerah, IPK, serta nilai TOEFL-nya.

b. Pemilihan dan Penambahan Data

Data yang akan digunakan dalam proses KDD adalah data tentang mahasiswa yang lulus Prodi Informatika dari Januari 2013 sampai Agustus 2017. Data tentang mahasiswa yang lulus tersebut diambil dari aplikasi repositori lulusan. Tidak semua data tentang mahasiswa yang lulus akan digunakan. Data yang akan digunakan adalah nama mahasiswa, jenis kelamin, asal daerah, IPK, nilai TOEFL, dan lama studi. Jenis kelamin, asal daerah, IPK, dan nilai TOEFL akan digunakan sebagai atribut dan lama studi akan digunakan sebagai kelas.

Dalam pembangunan aplikasi prediksi kelulusan mahasiswa, IPK menggambarkan performa akademik mahasiswa. Nilai TOEFL menggambarkan pemahaman mahasiswa dalam memahami literatur pembelajaran yang menggunakan bahasa Inggris. Asal daerah menggambarkan pengaruh faktor keluarga dan perbedaan kultur terhadap performa akademik mahasiswa. Keluarga dan kemandirian belajar merupakan faktor yang menentukan prestasi mahasiswa [4]. Selain itu fenomena *culture shock* seringkali terjadi pada mahasiswa perantauan. *Culture shock* tersebut dapat menimbulkan efek stres yang dapat mempengaruhi prestasi mahasiswa [5]. Sedangkan jenis kelamin menggambarkan pengaruh gender terhadap performa akademik mahasiswa. Dalam sebuah penelitian yang meneliti pengaruh gender dan motivasi belajar terhadap prestasi siswa, perempuan dinilai lebih berprestasi

Tabel 1. Tabel Transformasi Data

No.	Atribut	Nilai Atribut	Keterangan
1.	Asal daerah	Jateng	Provinsi bernilai Jawa Tengah
		Luar Jateng	Provinsi selain Jawa Tengah yang masih berada di Pulau Jawa
		Luar Jawa	Provinsi di luar Pulau Jawa
2.	IPK	Memuaskan	IPK kurang dari 2,75
		Sangat Memuaskan	IPK di antara 2,76 sampai dengan 3,50
		Dengan Pujian	IPK lebih dari 3,50
		Dasar	Nilai TOEFL kurang dari 420
3.	TOEFL	Menengah Bawah	Nilai TOEFL di antara 421 sampai 480
		Menengah Atas	Nilai TOEFL di antara 481 sampai 520
		Mahir	Nilai TOEFL lebih dari 520
4.	Lama studi	<5 tahun	lama studi kurang dari 5 tahun
		≥ 5 tahun	lama studi lebih dari sama dengan 5 tahun

daripada laki-laki. Hal ini dikarenakan perempuan lebih tekun dan rajin daripada laki-laki [6].

Jumlah data yang digunakan adalah 382 data dengan 212 data untuk kelas <5 tahun dan 170 data untuk kelas ≥ 5 tahun.

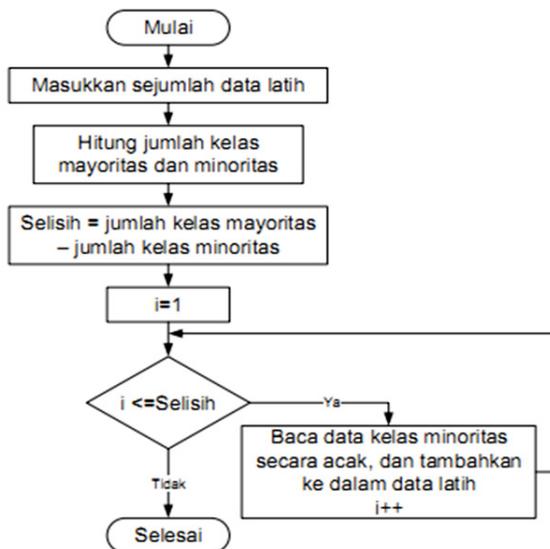
c. Pembersihan dan Pemrosesan Awal Data

Pembersihan data dilakukan untuk membersihkan *noise* pada data. Dalam penelitian ini proses pembersihan data dilakukan dalam aplikasi repositori lulusan. Aplikasi repositori lulusan telah diatur sedemikian rupa sehingga setiap data tentang mahasiswa yang lulus termasuk data nama, jenis kelamin, asal daerah, IPK, nilai TOEFL, dan lama studi yang disimpan tidak kosong dan konsisten.

d. Transformasi Data

Transformasi data dilakukan untuk mengubah data menjadi nilai dengan format tertentu. Dalam proses KDD ini akan digunakan data yang bersifat diskrit, oleh karena itu data yang bersifat kontinu akan diubah menjadi data diskrit. Selain itu ada data yang cakupannya terlalu luas dan akan mempengaruhi proses KDD sehingga perlu dikelompokkan menjadi beberapa kelompok kecil.

Data asal daerah merupakan data yang cakupannya luas. Asal daerah dalam data lulusan berisi nama provinsi asal dari masing-masing mahasiswa. Asal daerah akan dibagi menjadi 3 (tiga) kelompok, yaitu Jateng (Jawa Tengah), luar Jateng (luar Jawa Tengah), dan luar Jawa.



Gambar 1. Flowchart ROS

Pengelompokan IPK akan dibagi menjadi 3 kategori, yaitu tinggi, sangat memuaskan, dan rendah. Nilai per kelompok dibagi berdasarkan Peraturan Rektor Universitas Diponegoro Tahun 2012 Pasal 20. Untuk mahasiswa dengan IPK kurang dari 2,75 akan dikelompokkan sebagai IPK memuaskan, mahasiswa dengan IPK antara 2,76 sampai 3,50 akan dikelompokkan sebagai IPK sangat memuaskan, dan mahasiswa dengan IPK lebih dari 3,50 akan dikelompokkan sebagai IPK dengan pujian.

Dalam penelitian yang berjudul “*Reading-Writing Relationship in First and Second Language*”, Carson dkk mengelompokkan nilai TOEFL menjadi 4 kelas, yaitu dasar, menengah bawah, menengah atas, dan mahir. Nilai TOEFL kurang dari 420 termasuk dalam kelas dasar, nilai di antara 421 sampai 480 termasuk dalam kelas menengah bawah, nilai di antara 481 sampai 520 termasuk dalam kelas menengah atas, dan untuk nilai di atas 520 termasuk dalam kelas mahir [7].

Pada penelitian ini kelulusan seorang mahasiswa S1 Universitas Diponegoro akan dibagi menjadi dua kelompok, yaitu <5 tahun dan ≥ 5 tahun.

e. Data Mining

Untuk mencapai tujuan dari proses KDD akan digunakan metode pohon keputusan sebagai metode *data mining* dan algoritma C4.5 untuk membangun pohon keputusan. Untuk proses pemotongan pohon keputusan akan digunakan metode *error-based pruning*. Selain itu dikarenakan adanya *imbalance data*, data yang akan digunakan dalam proses pembuatan pohon keputusan akan ditangani dengan menggunakan algoritma *random over sampling* (ROS).

Tahap *data mining* dimulai dengan membagi data tentang mahasiswa yang lulus yang telah ditransformasi menjadi data latih dan data uji. Data latih akan digunakan dalam proses pembuatan pohon keputusan. Sedangkan data uji akan digunakan untuk mengukur kinerja dari pohon keputusan yang telah dibuat. Data latih yang akan digunakan dalam proses pembuatan pohon keputusan harus diseimbangkan terlebih dahulu untuk menghindari adanya kecenderungan terhadap kelas mayoritas dalam pohon yang dibuat. Proses untuk menyeimbangkan data

latih dilakukan dengan menggunakan metode *random over sampling* (ROS). Metode ROS dilakukan dengan menghitung selisih dari kelas mayoritas dan kelas minoritas. Kemudian dipilih satu data secara acak dari kelas minoritas. Data tersebut lalu ditambahkan ke dalam kelas minoritas. Penambahan data akan diulangi sampai jumlah data dalam kelas mayoritas dan kelas minoritas sama [8]. Metode ROS digambarkan dengan *flowchart* pada gambar 1.

Setelah data latih telah seimbang, data latih siap digunakan dalam proses membuat pohon keputusan dengan menggunakan algoritma C4.5. Algoritma C4.5 merupakan pengembangan dari algoritma ID3 yang ditemukan oleh J. Ross Quinlan pada tahun 1993 [9]. Algoritma C4.5 adalah sebagai berikut [10]:

- 1) Menghitung jumlah kasus total, jumlah kasus dengan keputusan <5 tahun, kasus dengan keputusan ≥ 5 tahun, dan *entropy* dari semua kasus dan kasus yang dibagi berdasarkan nilai atribut.

$$Entropy(S) = \sum_{j=1}^k -p_j \log_2 p_j \quad (1)$$

S = kumpulan data
k = banyaknya kelas dalam S
 p_j = probabilitas kelas C_j

Jika kasus total hanya memiliki satu kelas (<5 tahun atau ≥ 5 tahun), maka jadikan node sebagai daun dengan nilai kelas yang mayoritas.

- 2) Menghitung *information gain* untuk setiap atribut.

$$gain(A) = entropy(S) - \sum_{i=1}^k \frac{|S_i|}{|S|} \times entropy(S_i) \quad (2)$$

S = kumpulan data
A = atribut
 A_i = nilai atribut ke-i
 $|S_i|$ = jumlah data untuk A_i
 $|S|$ = jumlah data dalam S
k = jumlah nilai atribut

- 3) Menghitung *split info* untuk setiap atribut.

$$SplitInfo(S, A) = -\sum_{i=1}^n \frac{S_i}{S} \log_2 \frac{S_i}{S} \quad (3)$$

S = kumpulan data
A = atribut
 A_i = nilai atribut ke-i
n = jumlah nilai atribut
 S_i = jumlah data untuk A_i

- 4) Menghitung *gain ratio* untuk setiap atribut.

$$GainRatio(A) = \frac{gain(A)}{SplitInfo(S, A)} \quad (4)$$

A = atribut
S = kumpulan data

- 5) Memilih atribut dengan nilai *gain ratio* terbesar sebagai *node*.
- 6) Membagi data berdasarkan nilai atribut dari atribut terpilih. Kemudian menggunakannya untuk melakukan langkah selanjutnya.
- 7) Ulangi langkah 1 sampai 6 hingga seluruh atribut digunakan atau memenuhi suatu kondisi berhenti.

Untuk pemotongan pohon keputusan akan digunakan metode *error-based pruning*. *Error-based pruning* seringkali dideskripsikan sebagai *pessimistic error pruning* yang

ditambah dengan kemungkinan untuk mengganti *parent node* dengan *child* bernilai maksimum. Namun sebenarnya perhitungan estimasi *pessimistic error* dalam dua metode tersebut dilakukan dengan cara yang sama sekali berbeda [9]. Dalam metode *Error-Based Pruning* pemotongan pohon dilakukan dengan cara menilai setiap *node* bukan daun dari bawah pohon. Jika pengganti sub-pohon dengan daun akan membuat estimasi *error rate* lebih rendah, maka pohon akan dipotong. Setiap estimasi *error rate* untuk semua pohon yang termasuk di dalam sub-pohon ini akan terpengaruh. Karena *error rate* untuk keseluruhan pohon akan menurun seiring dengan menurunnya *error rate* dari sub-pohon, proses ini akan membuat sebuah pohon keputusan dengan *error rate* minimal, sehubungan dengan cara pemotongan pohon yang dilakukan [10]. Metode *error-based pruning* adalah sebagai berikut:

- 1) Menghitung jumlah kasus total, jumlah kasus dengan keputusan <5 tahun, dan kasus dengan keputusan ≥5 tahun dari sub pohon yang akan dihitung.
- 2) Menghitung nilai *f* untuk masing-masing *node* pada sub pohon. Nilai didapatkan dengan membagi jumlah kasus dengan keputusan kelas minoritas dengan jumlah kasus total.
- 3) Menghitung *error estimate* dari setiap *node* pada sub pohon.

$$e = \frac{f + \frac{z^2}{2N} + z \sqrt{\frac{f(1-f)}{N} + \frac{z^2}{4N^2}}}{1 + \frac{z^2}{N}} \quad (5)$$

e = estimasi *error rate*

f = kesalahan dalam data pelatihan

N = jumlah kasus pada *node*

z = *confidence limit*, dimana $z = z_{1 - (\alpha/2)}$ untuk *confidence level* α

- 4) Menghitung *error estimate* rata-rata untuk *node* anak sesuai dengan rasionya. Rasio untuk *node* anak dihitung dengan membagi jumlah kasus *node* anak dengan jumlah kasus *node* orang tua.
- 5) Membandingkan *error estimate* anak dan orang tua. Jika *error estimate* orang tua lebih kecil dari anak, maka pohon akan dipotong. Lalu *node* orang tua akan diubah menjadi daun dengan nilai kelas mayoritas. Sebaliknya jika *node* anak lebih kecil dari orang tua, pohon tidak dipotong.
- 6) Ulangi langkah 1 sampai 5 hingga seluruh sub pohon diperiksa.

3. Hasil

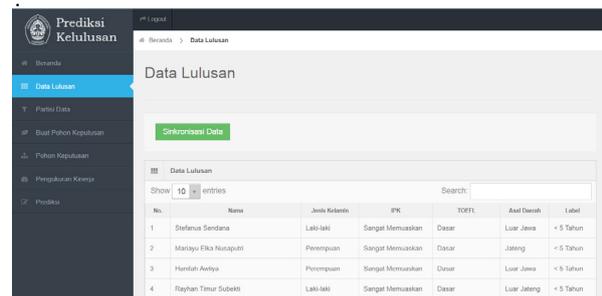
Pada bagian ini disajikan hasil penelitian yang berupa implementasi aplikasi prediksi kelulusan serta interpretasi dan evaluasinya.

a. Implementasi

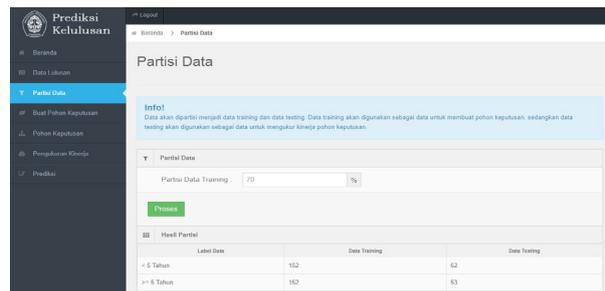
Implementasi program dilakukan berdasarkan kebutuhan fungsional yang tertera pada tabel 2. Untuk implementasi dari fungsional menampilkan data lulusan (SRS-PrediksiMhs-F-01) dapat dilihat pada gambar 2, berikutnya adalah implementasi dari fungsional melakukan dan menampilkan partisi data (SRS-PrediksiMhs-F-02) untuk memisahkan antara data *training* dengan data uji, disajikan pada gambar 3

Tabel 2. Kebutuhan fungsional

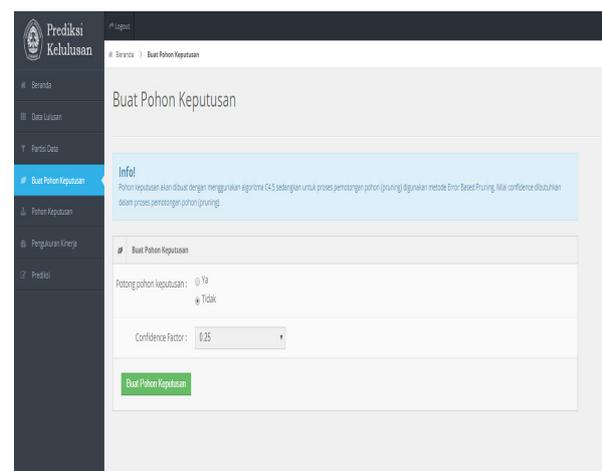
No.	SRS-ID	Deskripsi
1.	SRS-PrediksiMhs-F-01	Menampilkan data lulusan
2.	SRS-PrediksiMhs-F-02	Melakukan dan menampilkan partisi data
3.	SRS-PrediksiMhs-F-03	Membuat dan menampilkan pohon keputusan
4.	SRS-PrediksiMhs-F-04	Melakukan dan menampilkan hasil pengukuran kinerja
5.	SRS-PrediksiMhs-F-05	Mengidentifikasi kelulusan mahasiswa berdasarkan kriteria yang ditentukan



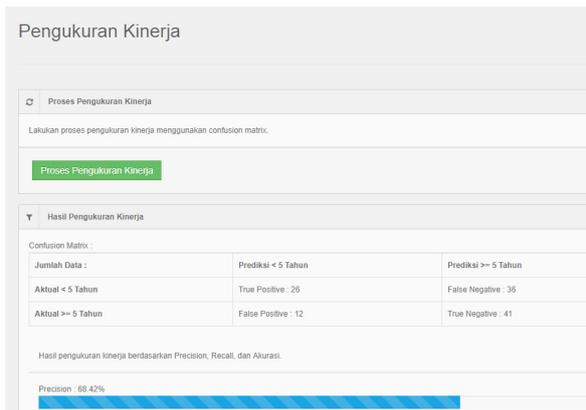
Gambar 2. Implementasi fungsional menampilkan data lulusan



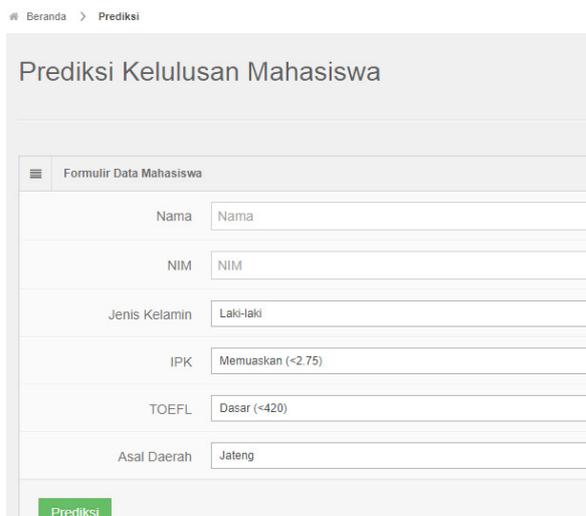
Gambar 3. Implementasi fungsional melakukan dan menampilkan partisi data



Gambar 4. Implementasi fungsional membuat dan menampilkan pohon keputusan



Gambar 5. Implementasi fungsional melakukan dan menampilkan hasil pengukuran kinerja



Gambar 6. Implementasi fungsional mengidentifikasi kelulusan mahasiswa berdasarkan kriteria yang ditentukan



Gambar 7. Hasil prediksi kelulusan mahasiswa

Implementasi dari fungsional membuat dan menampilkan pohon keputusan (SRS-PrediksiMhs-F-03) beserta pengaturan nilai *confidence factor* dapat dilihat pada gambar 4. Implementasi dari fungsional melakukan pengukuran kinerja dan menampilkan hasilnya (SRS-PrediksiMhs-F-04) disajikan pada gambar 5.

Yang terakhir adalah fitur mengidentifikasi kelulusan mahasiswa berdasarkan kriteria yang ditentukan (SRS-PrediksiMhs-F-05) dapat dilihat tampilan implementasinya pada gambar 6, yang hasilnya disajikan pada gambar 7.

Tabel 3. Detail pengukuran kinerja

Pengukuran ke-	Data Latih	Data Uji	Pruning	Nilai <i>confidence</i>
1	304	115	Tidak	-
2	304	115	Ya	0,25
3	304	115	Ya	0,4
4	308	115	Tidak	-
5	308	115	Ya	0,25
6	308	115	Ya	0,4
7	302	115	Tidak	-
8	302	115	Ya	0,25
9	302	115	Ya	0,4
10	298	115	Tidak	-
11	298	115	Ya	0,25
12	298	115	Ya	0,4
13	290	115	Tidak	-
14	290	115	Ya	0,25
15	290	115	Ya	0,4

Tabel 4. Confusion matrix pengukuran ke-1

	Prediksi <5 tahun	Prediksi ≥5 tahun
Aktual <5 tahun	True positives: 38	False negatives: 22
Aktual ≥5 tahun	False positives: 23	True negatives: 32

Tabel 5. Hasil pengukuran kinerja

Pengukuran ke-	Precision	Recall	Akurasi
1	62.3%	63.33%	60.86%
2	69.7%	38.33%	59.13%
3	69.7%	38.33%	59.13%
4	64%	55.17%	61.74%
5	66.07%	63.79%	65.22%
6	64.58%	53.45%	61.74%
7	60.66%	60.66%	58.26%
8	69.23%	59.02%	64.35%
9	69.23%	59.02%	64.35%
10	65.52%	60.32%	60.87%
11	83.87%	41.27%	63.48%
12	83.87%	41.27%	63.48%
13	67.19%	64.18%	60.87%
14	80%	41.79%	60%
15	66.13%	61.19%	59.13%

Tabel 6. Hasil rata-rata pengukuran kinerja

Pruning	Nilai <i>confidence</i>	Precision	Recall	Akurasi
Tidak	-	63.93%	60.73%	60.52%
Ya	0,25	73.77%	48.84%	62.44%
Ya	0,4	70.70%	50.65%	61.57%

b. Interpretasi

Pola yang dihasilkan dari proses *data mining* dapat ditampilkan dalam bentuk pohon keputusan dan *rules*. Berikut adalah contoh pohon keputusan yang dihasilkan:

```

IPK = Sangat Memuaskan = ≥5 Tahun
IPK = Dengan Pujian
| TOEFL = Dasar = <5 Tahun
| TOEFL = Menengah Bawah = <5 Tahun
| TOEFL = Menengah Atas = <5 Tahun
| TOEFL = Mahir = ≥5 Tahun
IPK = Memuaskan = ≥5 Tahun
  
```

Dari pohon keputusan tersebut dapat diambil *rules* sebagai berikut:

1. IF IPK sangat memuaskan THEN ≥5 tahun
2. IF IPK dengan pujian AND TOEFL dasar THEN <5 tahun
3. IF IPK dengan pujian AND TOEFL menengah bawah THEN <5 tahun
4. IF IPK dengan pujian AND TOEFL menengah atas THEN <5 tahun
5. IF IPK dengan pujian AND TOEFL mahir AND asal daerah Jateng AND jenis kelamin perempuan THEN <5 tahun
6. IF IPK dengan pujian AND TOEFL mahir THEN ≥5 tahun
7. IF IPK memuaskan THEN ≥5 tahun

c. Pengukuran Kinerja Pohon Keputusan

Pengukuran kinerja dilakukan untuk mengevaluasi pohon keputusan yang telah dibangun sebelumnya. Pengukuran kinerja dilakukan dengan menggunakan *confusion matrix*. *Confusion matrix* didapatkan dengan melakukan perbandingan hasil prediksi dari aplikasi dan hasil sebenarnya. Untuk mengetahui kinerja dari pohon keputusan akan digunakan *precision*, *recall*, dan akurasi.

Proses pengukuran kinerja akan dilakukan sebanyak 15 kali terhadap 5 data uji berbeda. Lima pengukuran kinerja akan dilakukan terhadap pohon keputusan tanpa pemotongan dan sisanya dilakukan terhadap pohon keputusan yang telah dipotong. Untuk pohon keputusan dengan pemotongan, proses pemotongan pohon dilakukan dengan nilai *confidence* sebesar 0,25 dan 0,4. Sedangkan data uji yang digunakan dalam proses pengukuran kinerja adalah sebesar 30% dari data tentang mahasiswa yang lulus yang telah di transformasi. Detail dari pengukuran kinerja yang akan dilakukan dapat dilihat pada tabel 3.

Untuk melakukan pengukuran kinerja pohon keputusan, langkah pertama yaitu hitung *true positives*, *true negatives*, *false positives*, dan *false negatives* dari setiap pengukuran, kemudian masukkan hasilnya ke dalam *confusion matrix*.

Langkah kedua, hitung *precision*, *recall*, dan akurasi. Akurasi didefinisikan sebagai tingkat kedekatan antara nilai hasil prediksi dengan nilai aktual. *Precision* didefinisikan sebagai pengukuran ketepatan. Jika data diprediksi positif, seberapa seringkah data prediksi tersebut benar. Sedangkan *recall* didefinisikan sebagai pengukuran kelengkapan. Dari jumlah data sebenarnya yang bernilai positif, sebanyak apakah data yang diprediksi positif [11].

$$Precision = TP / (TP+FP) \quad (6)$$

$$Recall = TP / (TP+FN) \quad (7)$$

$$Akurasi = (TP+TN) / N \quad (8)$$

TP = nilai *true positives*

TN = nilai *true negatives*

FP = nilai *false positives*

FN = nilai *false negatives*

N = jumlah data

Kemudian ubah nilai *precision*, *recall*, dan akurasi menjadi nilai persentase. Hasil dari pengukuran yang telah dilakukan dapat dilihat pada tabel 5.

Hasil pada tabel 5 kemudian dirata-rata berdasarkan pohon keputusan tanpa pemotongan, pohon keputusan yang dipotong dengan nilai *confidence* 0,25, dan pohon keputusan yang dipotong dengan nilai *confidence* 0,4.

Pengukuran kinerja yang dilakukan terhadap pohon keputusan tanpa pemotongan menghasilkan nilai *precision* sebesar 63,93%, nilai *recall* sebesar 60,73%, dan nilai akurasi sebesar 60,52%. Sedangkan untuk pohon keputusan yang dipotong dengan nilai *confidence* 0,25 menghasilkan nilai *precision* sebesar 73,77%, nilai *recall* sebesar 48,84%, dan nilai akurasi sebesar 62,44%. Pengukuran kinerja untuk pohon keputusan yang dipotong dengan nilai *confidence* 0,4 menghasilkan nilai *precision* sebesar 70,70%, nilai *recall* sebesar 50,65%, dan nilai akurasi sebesar 61,57%. Dari hasil tersebut dapat diketahui bahwa pemotongan pohon keputusan dengan menggunakan metode *error-based pruning* dapat meningkatkan akurasi. Pemotongan dengan menggunakan nilai *confidence* sebesar 0,25 meningkatkan akurasi lebih baik daripada nilai *confidence* 0,4.

d. Mengolah Pengetahuan

Hasil dari proses KDD dalam penelitian ini adalah prediksi apakah seorang mahasiswa akan lulus <5 tahun atau ≥5 tahun. Hasil tersebut langsung digunakan dengan diperlihatkan secara langsung kepada pengguna setelah proses prediksi dilakukan.

4. Kesimpulan

Kesimpulan yang dapat diambil dari hasil penelitian ini adalah sebagai berikut:

1. Dari hasil penerapan algoritma C4.5 dalam prediksi kelulusan mahasiswa Prodi Informatika dapat disimpulkan bahwa atribut yang paling dominan dalam kelulusan mahasiswa adalah IPK, kedua adalah TOEFL, ketiga adalah asal daerah, dan yang terakhir adalah jenis kelamin.
2. Pemotongan pohon keputusan pada algoritma C4.5 dapat meningkatkan akurasi. Pohon tanpa pemotongan menghasilkan nilai rata-rata *precision* 63.93%, *recall* 60.73%, dan akurasi 60.52%. Sedangkan pohon keputusan yang dipotong dengan menggunakan metode *error-based pruning* dengan menggunakan nilai *confidence* 0,4 menghasilkan *precision* 70.70%, *recall* 50.65%, dan akurasi 61.57%. Pohon yang dipotong dengan menggunakan nilai *confidence* 0,25 menghasilkan *precision* 73.77%, *recall* 48.84%, dan akurasi 62.44%. Dari hasil tersebut dapat disimpulkan bahwa penggunaan nilai *confidence* 0,25 meningkatkan akurasi lebih baik daripada nilai *confidence* 0,4.

5. Daftar Pustaka

- [1] S. D. Jadhav and H. Channe, "Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques," *International Journal of Science and Research (IJSR)*, vol. 5, no. 1, pp. 1842-1845, January 2016.
- [2] M. Mohankumar, S. Amuthakkani and G. Jeyamala, "Comparative Analysis of Decision Tree Algorithms for The Prediction of Eligibility of A Man for Availing Bank Loan," *International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)*, vol. 2, no. 15, pp. 360-366, 2016.
- [3] D. H. Kamagi, "Implementasi Data Mining dengan Algoritma C4.5 untuk Memprediksi Tingkat Kelulusan Mahasiswa," *ULTIMATICS*, vol. VI, no. 1, pp. 15-20, Juni 2014.
- [4] V. Anggresta, "Analisis Faktor-faktor yang Mempengaruhi Belajar Mahasiswa Fakultas Ekonomi Universitas Negeri Padang," *Journal of Economic and Economic Education*, vol. 4, pp. 19-29, 2015.
- [5] M. Devinta, "Fenomena Culture Shock (Gegar Budaya) pada Mahasiswa Perantauan di Yogyakarta," *Jurnal Pendidikan Sosiologi*, 2015.
- [6] Y. Kusnia, "Pengaruh Karakteristik Gender dan Motivasi Belajar terhadap Prestasi Belajar Matematika Siswa Kelas X IPA 1 di MAN 2 Semarang," 2017.
- [7] J. E. Carson, P. L. Carrell, S. Silberstein, B. Kroll and P. A. Kuehn, "Reading-Writing Relationships in First and Second Language," *TESOL Quarterly*, vol. 24, pp. 245-266, 1990.
- [8] A. Saifudin and R. S. Wahono, "Pendekatan Level Data untuk Menangani Ketidakseimbangan Kelas pada Prediksi Cacat Software," *Journal of Software Engineering*, vol. 1, pp. 76-85, 2015.
- [9] K. Grabczewski, *Meta-Learning in Decision Tree Induction*, Springer, 2014.
- [10] J. R. Quinlan, *C4.5: Programs for Machine Learning*, San Mateo: Morgan Kaufman, 1993.
- [11] J. Han, M. Kamber and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed., Waltham: Elsevier Inc., 2014.

Perancangan Aplikasi *Chatting* Berbasis *Web* di PT. Pura Barutama Kudus Menggunakan Socket.IO dan *Framework Foundation*

Ramos Somya*

Program Studi S1 Teknik Informatika, Fakultas Teknologi Informasi

Universitas Kristen Satya Wacana

Salatiga

*ramos.somya@uksw.edu

Abstrak-Aplikasi *chatting* merupakan aplikasi yang penting pada sebuah perusahaan besar karena dapat dipastikan antara divisi satu dengan divisi yang lain berada di lokasi yang berjauhan. EDP Keuangan PT. Pura Barutama Kudus bertugas dalam pembuatan program aplikasi yang terkait dengan pengolahan data keuangan dan faktur di semua unit di PT. Pura Barutama. Aplikasi keuangan yang telah ada akan selalu berkembang sesuai dengan kebutuhan *user*. Kendala yang dihadapi adalah aktivitas komunikasi dan penyebaran informasi pada PT. Pura Barutama dilakukan dengan menggunakan layanan *email* maupun aplikasi *messenger*. Untuk menggunakan layanan tersebut, perangkat yang digunakan harus selalu terkoneksi dengan internet. Apabila internet mati maka komunikasi tidak dapat dilakukan. Jalan satu-satunya untuk menyebarkan informasi adalah dengan menelepon satu per satu ke tiap-tiap unit. Hal ini menimbulkan ketidakefisienan pekerjaan dan produktivitas perusahaan menjadi terganggu. Pada penelitian ini dilakukan perancangan aplikasi *chatting* dengan menggunakan *Framework CodeIgniter*, Socket.IO dan *Framework Foundation* untuk mengatasi masalah yang ada. Hasil dari penelitian ini adalah aplikasi *chatting* berbasis *web* yang dibangun menggunakan *Framework CodeIgniter*, Socket.IO dan *Framework Foundation* terbukti dapat membantu *user* untuk saling berkomunikasi atau melakukan *attach file*, serta mempermudah melakukan penyebaran informasi atau pengumuman baik dalam kondisi terhubung internet maupun tidak.

Kata Kunci: Aplikasi Chatting, Web, Socket.IO, Framework Foundation, Framework CodeIgniter

1. Pendahuluan

Aplikasi *chatting* merupakan aplikasi yang penting pada sebuah perusahaan besar karena dapat dipastikan antara divisi satu dan divisi yang lainnya berada di tempat yang berjauhan. Divisi keuangan PT. Pura Barutama selalu mengontrol dan berkomunikasi dengan seluruh unit yang ada di PT. Pura Barutama. Penggunaan aplikasi *chatting* akan meningkatkan produktivitas dan efisiensi kerja. Misalnya sebagai sarana berkomunikasi, *attach file* dan sebagai media pengumuman.

EDP Keuangan adalah salah satu sub divisi dari divisi keuangan di PT. Pura Barutama selalu berinteraksi dengan berbagai macam divisi yang ada. Selain itu EDP Keuangan bertugas dalam pembuatan program yang terkait dengan pengolahan data keuangan, faktur di semua unit PT. Pura Barutama. Aplikasi keuangan yang telah ada akan selalu berkembang sesuai kebutuhan *user*.

Aktivitas komunikasi dan penyebaran informasi pada PT. Pura Barutama dilakukan dengan menggunakan layanan *email* maupun aplikasi *messenger*, untuk menggunakan layanan tersebut, perangkat yang digunakan harus selalu terkoneksi dengan internet. Apabila internet mati maka komunikasi tidak dapat dilakukan. Jalan satu-satunya untuk menyebarkan informasi adalah dengan menelepon satu per satu ke tiap-tiap unit. Hal ini menimbulkan

ketidakefisienan pekerjaan dan produktivitas menjadi terganggu.

Berdasarkan permasalahan tersebut, maka dapat diketahui rumusan masalah dalam penelitian ini adalah bagaimana membuat aplikasi *chatting* berbasis *web* menggunakan *Framework CodeIgniter*, Socket.IO dan *Framework Foundation* dengan memanfaatkan jaringan komputer lokal yang ada di PT. Pura Barutama. Aplikasi *chatting* tersebut hanya membutuhkan sebuah *browser* yang digunakan untuk mengakses aplikasi *chatting* selama perangkat terkoneksi dengan jaringan lokal baik menggunakan kabel maupun *wireless* tanpa terkoneksi dengan internet. Selain itu, aplikasi berbasis *web* tergolong ringan, sehingga spesifikasi komputer yang digunakan oleh *user* tidak berat.

Aplikasi *chatting* dibangun menggunakan *Framework CodeIgniter* (CI) dan Socket.IO sebagai *back-end*. *Framework CodeIgniter* mendukung konsep *Model View Controller* (MVC) sehingga dalam pengembangan aplikasi *chatting* akan menjadi lebih terstruktur dan terorganisasi. Sedangkan untuk *interface* (*front-end*) aplikasi *chatting* menggunakan *Framework Foundation* untuk membuat aplikasi *chatting* ini menjadi *responsive* dan untuk penyimpanan data menggunakan *database* Oracle karena menyesuaikan dengan *database* yang digunakan PT. Pura Barutama.

Mengingat begitu luasnya ruang lingkup pembuatan aplikasi *chatting* dan terbatasnya kemampuan dan identifikasi maka diperlukan batasan agar sistem yang dibangun tidak menyimpang dari permasalahan dalam sistem aplikasi *chatting*. Adapun batasan masalah yang dilakukan adalah aplikasi hanya dapat dipergunakan untuk perorangan (*Private Chat*), dikarenakan aplikasi *chatting* ini bersifat rahasia. Selain digunakan untuk melakukan *private chat*, aplikasi ini juga dapat memberikan informasi berupa pengumuman kepada *user* maupun unit yang lain apabila ada pembaruan aplikasi yang dibuat oleh EDP Keuangan PT. Pura Barutama.

Diharapkan dengan dibangunnya aplikasi ini, dapat mempermudah para penggunanya untuk berkomunikasi dan bertukar informasi. Sehingga dapat meningkatkan produktivitas kerja dan mengefisienkan waktu

2. Kajian Pustaka

a. Penelitian Terdahulu

Terdapat beberapa penelitian terdahulu yang digunakan sebagai acuan dalam penelitian ini. Penelitian pertama menghasilkan aplikasi yang dapat melakukan pembuatan suatu *room* yang dapat dilakukan oleh semua pengguna baik itu pengguna telepon seluler maupun pengguna komputer. Apabila melakukan mengirim dengan ukuran *file* yang besar maka proses pengiriman akan menjadi lama [1].

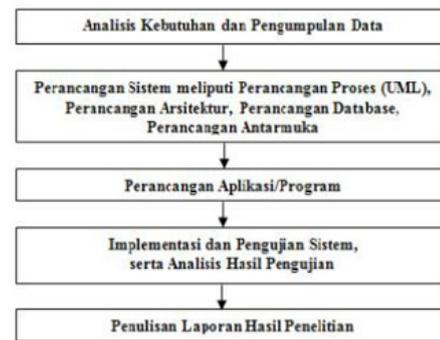
Penelitian kedua menghasilkan aplikasi yang dibangun dengan menggunakan Visual Basic 6.0. Aplikasi yang dilengkapi dengan menu *send file* untuk mengirimkan *text* yang mempermudah komunikasi antar pengguna yang berada dalam sebuah jaringan. Aplikasi *chatting* yang dikembangkan tidak adanya pemisahan form antara *private* dengan *group*. Model pengiriman *text* melalui menu *send file* harus memasukkan alamat IP tujuan pengguna sehingga dinilai tidak efektif dan efisien. Program aplikasi hanya dapat berjalan pada sistem operasi Windows [2].

Penelitian ketiga mengimplementasi Algoritma Rijndael 128 pada aplikasi *chatting* berbasis HTML5 WebSocket. Pada penelitian tersebut menghasilkan aplikasi *group chatting*. Pesan yang dikirim akan dienkripsi dengan menggunakan algoritma Rijndael 128, sehingga pesan menjadi aman [3].

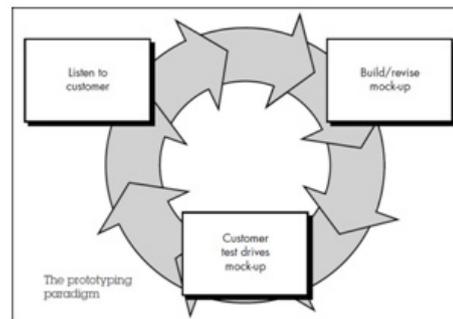
Berdasarkan penelitian yang telah dilakukan sebelumnya yang menjadi pembeda antara penelitian ini dan penelitian terdahulu adalah aplikasi *chatting* dibuat dengan menggunakan *Framework* CodeIgniter, Socket.IO dan *Framework* Foundation. Penggunaan *Framework* CodeIgniter penulisan kode akan menjadi terstruktur dan terorganisir karena *Framework* CodeIgniter menerapkan konsep MVC (*Model View Controller*). Socket.IO memungkinkan pembuatan aplikasi komunikasi *realtime* antara *client* dan *server*. Penggunaan *Framework* Foundation membuat halaman *web* menjadi *responsive* sehingga dapat menyesuaikan di perangkat manapun saat aplikasi diakses.

b. Chatting

Chatting adalah pecakapan dua orang atau lebih secara *real time* melalui komputer yang terhubung dengan jaringan. Layanan untuk *chatting* di internet antara lain Yahoo, Skype, mIRC, Windows Live Messenger. Adanya layanan *chat* memungkinkan untuk dapat berkomunikasi melalui internet dengan orang-orang yang berada di seluruh dunia [4].



Gambar 1. Tahapan Penelitian



Gambar 2. Model prototyping [10]

c. Framework

Framework adalah kumpulan perintah atau fungsi dasar yang membentuk aturan-aturan tertentu dan saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi *website*, *programmer* harus mengikuti aturan dari *framework* tersebut. CodeIgniter dilengkapi dengan berbagai pustaka siap pakai untuk berbagai kebutuhan. Misalnya saja koneksi *database*, email, *session* dan *cookies*, keamanan, manipulasi gambar dan banyak lagi sehingga mempermudah pekerjaan [5].

d. Socket.IO

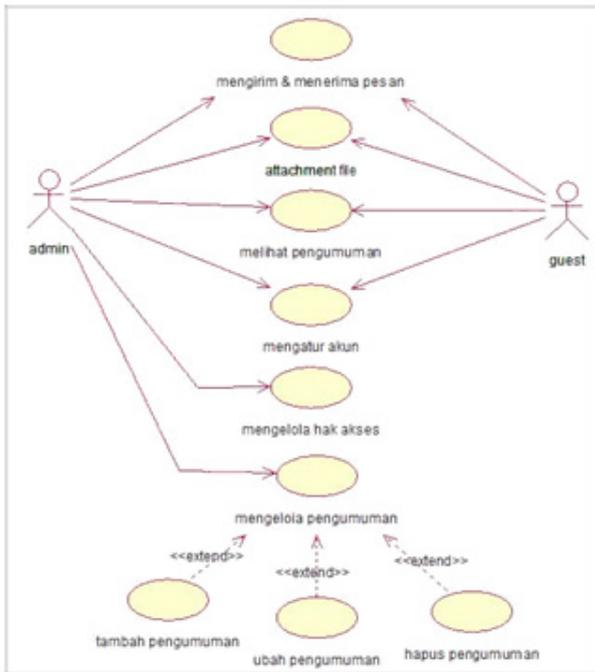
Socket.IO adalah lapisan komunikasi berbasis *event* untuk aplikasi *web realtime*, yang dibangun di atas Engine.IO. Hal ini memungkinkan pengembang untuk mengirim dan menerima data tanpa khawatir tentang kompatibilitas *browser* yang berbeda [6].

e. Framework Foundation

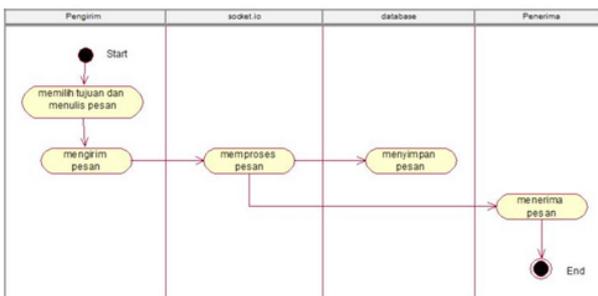
Framework Foundation adalah *framework* yang dibangun dengan HTML, CSS, dan Javascript, sebagai komponen utama dari *web*. *Framework* Foundation menggunakan teknologi JQuery, HTML 5 dan Normalizer [7]. *Framework* Foundation digunakan untuk membuat *web* dapat menyesuaikan resolusi layar yang digunakan.

f. Basis Data Oracle

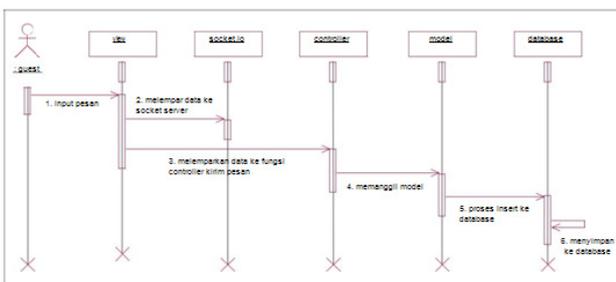
Oracle adalah sistem basis data yang memiliki banyak fitur yang memungkinkan administrator basis data dapat mengelola data secara lebih akurat sehingga Oracle lebih sesuai digunakan sebagai sistem basis data untuk aplikasi yang berukuran besar dan kompleks [8].



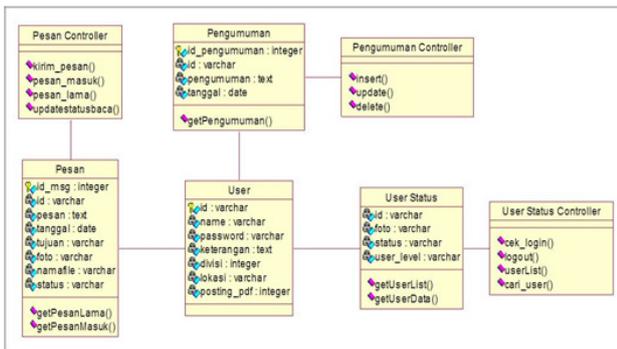
Gambar 3. Diagram Use Case



Gambar 4. Diagram Activity Mengirim Pesan



Gambar 5. Diagram Sequence Proses Kirim Pesan



Gambar 6. Diagram Class

3. Perancangan Sistem

Pada penelitian ini dilakukan beberapa tahapan yang saling berkaitan dengan tahapan selanjutnya, yaitu: 1) Analisis kebutuhan dan pengumpulan data yang diperlukan. 2) Perancangan sistem. 3) Perancangan aplikasi/program. 4) Implementasi dan pengujian sistem, serta analisis hasil pengujian. 5) Penulisan laporan hasil penelitian [9]. Tahapan-tahapan yang dilakukan dalam penelitian ini dapat dilihat pada Gambar 1.

Berdasarkan Gambar 1 dapat dijelaskan tahapan penelitian yang dilakukan adalah sebagai berikut: 1) Tahapan pertama: analisis dan pengumpulan data, di mana peneliti melakukan wawancara dengan karyawan EDP Keuangan tentang aplikasi yang akan dibuat. Berdasarkan hasil wawancara dengan karyawan EDP Keuangan bahwa selama ini belum ada media yang menghubungkan unit satu ke unit yang lainnya dalam menyebarkan informasi atau *attach file* tanpa menggunakan internet. Selama ini hanya menggunakan telepon dan *email* untuk berkomunikasi. *Email* sendiri memiliki kekurangan yaitu membutuhkan koneksi internet. Jika koneksi internet mati, maka tidak bisa terjadi komunikasi. Jika melalui telepon akan terasa tidak efisien karena karyawan EDP Keuangan harus menghubungi satu per satu unit yang banyak di PT. Pura Barutama jika terjadi perubahan program. Sehingga dibutuhkan aplikasi yang dapat menyediakan fasilitas *chatting*, *attach file*, dan penyebaran pengumuman dengan memanfaatkan jaringan lokal. Tahap kedua, ketiga, dan keempat dilakukan perancangan aplikasi *chatting* menggunakan metode Prototype. Sedangkan tahap kelima dilakukan penulisan artikel ilmiah atau laporan penelitian.

Metode perancangan yang dipakai dalam pembuatan aplikasi *chatting* adalah metode Prototyping. Metode Prototyping adalah metode dalam pengembangan rekayasa *software* yang bertahap dan berulang serta mementingkan sisi *user* sistem. Penggunaan metode Prototyping, pengembang dan karyawan EDP Keuangan dapat saling berinteraksi selama proses pembuatan sistem sampai aplikasi sesuai dengan kebutuhan pengguna.

Tahap pengumpulan kebutuhan dilakukan untuk mengetahui permasalahan dan kebutuhan sistem. Pada tahap ini juga dilakukan pencarian data yang dibutuhkan oleh sistem. Agar aplikasi *chatting* yang dibangun dapat memenuhi kebutuhan pengguna. Analisis kebutuhan sistem dilakukan bersama dengan karyawan EDP Keuangan. Berdasarkan analisis kebutuhan sistem yang dilakukan bahwa selain untuk melakukan *chatting* dengan user lain, dibutuhkan juga fasilitas untuk *attach file*, dan melakukan pengumuman. Pada aplikasi *chatting* yang dibangun dibagi menjadi dua hak akses yaitu administrator dan *user* biasa. Kebutuhan administrator mencakup: *chatting* dengan pengguna lain dengan fasilitas *attach file*, *posting* pengumuman, hapus pengumuman, mengubah pengumuman jika terjadi kesalahan, mengelola *user*, mengelola *user* di sini maksudnya adalah memberikan hak akses administrator kepada *user* biasa jika *user* tersebut ini *posting* pengumuman. Sedangkan kebutuhan *user* biasa hanya melakukan *chatting* dengan pengguna lain dengan fasilitas *attach file*. Data *user* sendiri menggunakan data yang sudah dimiliki oleh PT Pura Barutama, sehingga pada aplikasi *chatting* ini tidak perlu menambahkan data *user* baru, karena diambil dari *database* yang sudah ada.

Analisis kebutuhan *hardware* dan spesifikasi *software* yang digunakan dalam membangun aplikasi *chatting* ini yaitu: analisis perangkat keras yang akan digunakan adalah Prosesor Intel Core i3, 2.3 GHz, RAM 6 GB dan Harddisk 500 GB. Sedangkan perangkat lunak yang digunakan adalah sistem Operasi Windows 7 Ultimate, Sublime sebagai *text editor*, *web server* (Apache), *web browser* (dalam penelitian ini menggunakan Google Chrome), Oracle sebagai *database*, dan Rational Rose untuk membuat UML diagram sistem.

Sebelum dilakukan pengkodean, dilakukan perancangan UML diagram untuk memvisualisasikan alur proses dan kebutuhan data. UML dibuat dalam diagram Use Case, diagram Activity, diagram Sequence dan diagram Class yang akan dijelaskan satu per satu.

Gambar 3 menunjukkan diagram Use Case dari aplikasi *chatting*. Aplikasi yang dibuat dibedakan menjadi dua jenis *user* yaitu administrator dan *guest* (*user* biasa). Aktor dengan hak akses admin dapat mengirimkan dan menerima pesan, *attach file*, melihat pengumuman, mengelola data *user* dengan memberikan hak akses administrator kepada *user* biasa, *memposting* pengumuman, menghapus pengumuman, mengubah pengumuman jika terjadi kesalahan. Sedangkan aktor dengan hak akses *guest* (*user* biasa) hanya dapat mengirimkan dan menerima pesan, *attach file*, melihat pengumuman, dan mengatur akunnya sendiri.

Gambar 4 menunjukkan diagram Activity ketika pengirim mengirimkan pesan. Hal pertama yang dilakukan adalah pengirim menentukan tujuan dan pesan yang akan dikirim. Kemudian data yang sudah diinputkan akan diproses melalui Socket kemudian ditampilkan pada penerima. Data pesan yang dikirim akan disimpan ke dalam *database*, sehingga histori percakapan masih bisa dilihat.

Gambar 5 merupakan diagram Sequence dari proses pengiriman pesan. Pertama *user* menginputkan pesan pada halaman *view*. Kemudian data diproses oleh *server* socket dan data menjadi parameter di *controller* kirim pesan. Setelah itu *controller* akan memanggil fungsi pada model dengan parameter yang dikirimkan *controller* kemudian model akan berhubungan dengan *database* untuk menyimpan data.

Gambar 6 merupakan diagram *Class* dari aplikasi *chatting* yang menunjukkan model dan *controller*. Pada aplikasi *chatting* dibuat dalam 4 model *class* yaitu *user class*, *user status class*, *pesan class* dan *pengumuman class*. Model *class* ini yang nantinya akan berhubungan dengan *database*. Sedangkan *controller class* akan menghubungkan antara *view* dan model *class*. Pada aplikasi *chatting* dibuat dalam 3 *controller class* yaitu *user status controller*, *pesan controller*, dan *pengumuman controller*.

Perulangan dari proses pada metode Prototyping terus berlangsung hingga semua kebutuhan terpenuhi. Proses evaluasi dilakukan sebanyak 3 kali, evaluasi pertama adalah menunjukkan kepada karyawan EDP Keuangan bagaimana aplikasi bekerja. Evaluasi kedua dilakukan penambahan fitur untuk *user* yang memiliki hak akses sebagai administrator agar dapat melakukan *posting* pengumuman, dan setiap pesan yang masuk mendapatkan notifikasi. Evaluasi ketiga dilakukan perbaikan beberapa fungsi yang masih memiliki *bug* di dalamnya.

Kode Program 1. Konfigurasi NodeJS

```
1.  {
2.  "name": "puramessenger",
3.  "version": "1.0.0",
4.  "description": "puramessenger",
5.  "main": "server.js",
6.  "scripts": {
7.  "test": "echo \"Error: no test specified\" &&
  exit 1"
8.  },
9.  "author": "ridvandani672012078",
10. "license": "ISC",
11. "dependencies":{
12. "Socket.IO":"*",
13. "express":"*"
14. }
15. }
```

Kode Program 2. Konfigurasi Server

```
1.  var express = require('express');
2.  var app = express();
3.  var server =
  require('http').createServer(app);
4.  var io =
  require('Socket.IO').listen(server);
5.  var port = process.env.PORT || 3001;
6.  var users = {};
7.  server.listen(port, function(){
8.  console.log("Server bejalan pada port %d",
  port);
9.  });
```

Kode Program 3. Konfigurasi Koneksi Database Framework CodeIgniter

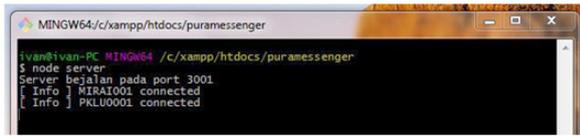
```
1.  $db['default']['hostname'] = 'localhost/XE';
2.  $db['default']['username'] = 'pura_m';
3.  $db['default']['password'] = 'pura_m';
4.  $db['default']['database'] = '';
5.  $db['default']['dbdriver'] = 'oci8';
6.  $db['default']['dbprefix'] = '';
7.  $db['default']['pconnect'] = TRUE;
8.  $db['default']['db_debug'] = TRUE;
9.  $db['default']['cache_on'] = FALSE;
10. $db['default']['cachedir'] = '';
11. $db['default']['char_set'] = 'utf8';
12. $db['default']['dbcollat'] =
  'utf8_general_ci';
13. $db['default']['swap_pre'] = '';
14. $db['default']['autoinit'] = TRUE;
15. $db['default']['stricton'] = FALSE;
```

Kode Program 4. Konfigurasi Route Framework CodeIgniter

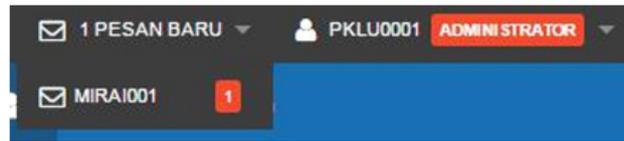
```
1.  $route['default_controller'] = "user_con";
2.  $route['404_override'] = '';
3.  $route['translate_uri_dashes'] = FALSE;
```

Kode Program 5. Menampilkan *User* yang *Online* atau *Offline*

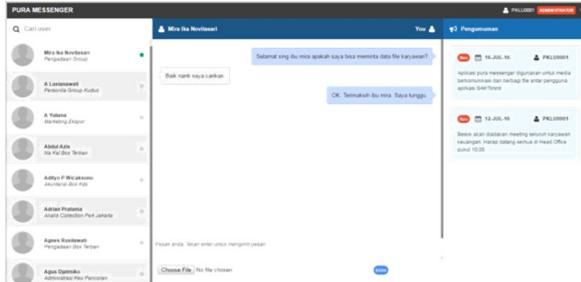
```
1.  function cek_userlist() {
2.  $.ajax({
3.  url: "<?php echo
  base_url('index.php/user_con/userList');?>",
4.  cache: false,
5.  success: function(msg) {
6.  $('#userList').load("<?php echo
  base_url('index.php/user_con/userList');?>");
7.  });
8.  });
9.  var waktu = setTimeout("cek_userlist()",
  10000);
10. }
```



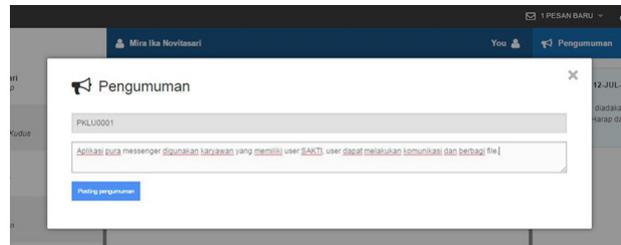
Gambar 7. Respon *Server* Ketika *User Log in*



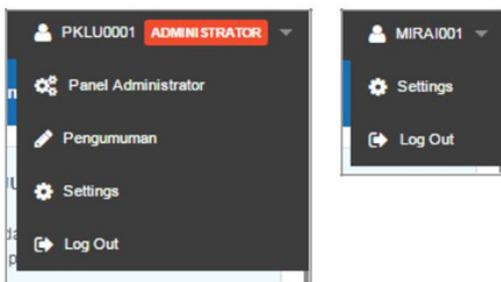
Gambar 12. Notifikasi Pesan yang Belum Dibaca



Gambar 8. Halaman Utama



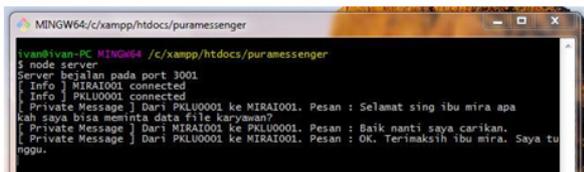
Gambar 13. Form *Posting* Pengumuman



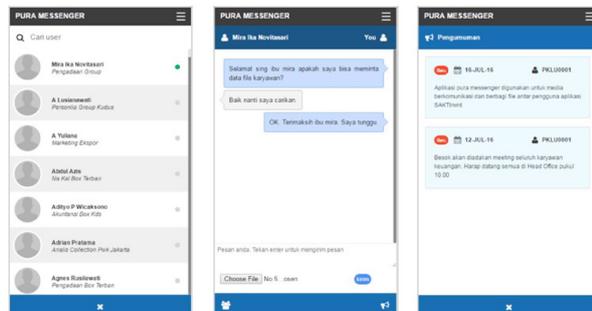
Gambar 9. Perbedaan Hak Akses Administrator dan *User* Biasa



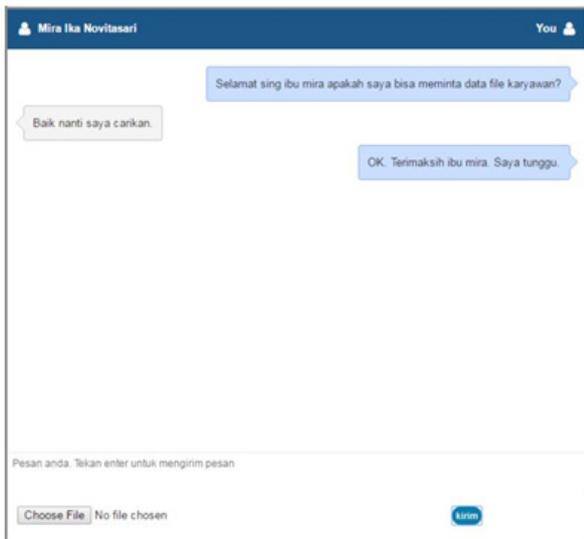
Gambar 14. Daftar Pengumuman



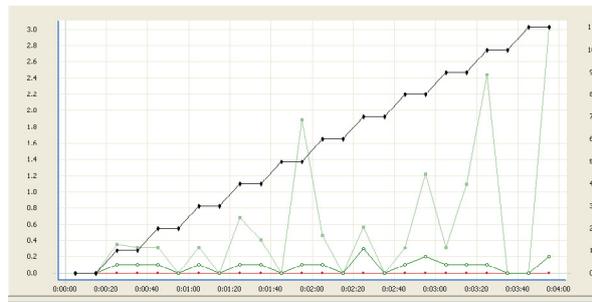
Gambar 10. Respon *Server* Ketika *User Mengirim Pesan*



Gambar 15. Tampilan *Responsive*



Gambar 11. Proses *Chatting*



Gambar 16. Grafik *Performance* Aplikasi

Kode Program 6. Mengirim Pesan Melalui Socket

```

1. $('#form-kirim-pesan').submit(function(e) {
2.   e.preventDefault();
3.   var url = $(this).attr('action');
4.   var data = new FormData(this);
5.   var pesan = '/w ' + $inp_tujuan.val() + " " +
     $inp_pesan.val();
6.   var filename =
     $('input[type=file]').val().replace(/C:\\fakep
     ath\\/i, '');
7.   var pData = {pesan: pesan, file: filename,
     gambar: $inp_gambar.val()};
8.
9.   Socket.emit('send message', pData);
10.  ...
11. });

```

Kode Program 7. Media Query

```

1. @media (max-width: 700px){
2.   .userListContainer {display:none;}
3.   .announcementContainer {display: none;}
4.   .chatbox{height: 67% !important;}
5.   }

```

4. Hasil dan Pembahasan

Sebelum melakukan pengkodean. Hal pertama yang harus dilakukan adalah meng-*install* NodeJS. Tujuan dari instalasi NodeJS karena Socket.IO berjalan di atas NodeJS sebagai modul. Kemudian dilakukan inisialisasi *project* NodeJS dengan perintah *npm init*. Perintah ini akan menghasilkan *file package.json* yang terlihat pada Kode Program 1.

Kode Program 1 menunjukkan deskripsi mengenai aplikasi yang dibuat. Kemudian melakukan penambahan *dependencies* untuk menambahkan Socket.IO dan Express. Setelah konfigurasi *package.json*, selanjutnya penginstallan Socket.IO dan Express dengan menjalankan perintah *npm install* pada *command line*. Setelah proses instalasi, selanjutnya membuat *server* untuk aplikasi *chatting* yang ditunjukkan pada Kode Program 2.

Kode Program 2 menunjukkan inisialisasi Socket dan *port*. Pada baris 5 diinisialisasikan bahwa aplikasi *chatting* akan berjalan pada *port* 3001. Selanjutnya dilakukan konfigurasi pada *Framework* CodeIgniter agar aplikasi dapat terhubung dengan *database* yang ditunjukkan pada Kode Program 3.

Kode Program 3 menunjukan konfigurasi *database* yang berisi *hostname*, *username*, *password*, dan *driver*. Pada penelitian ini *database* yang digunakan adalah Oracle, maka *driver* yang digunakan adalah OCI8. Sedangkan untuk pengaturan *route* terlihat pada Kode Program 4.

Kode Program 4 menunjukkan *controller* mana yang akan dipanggil pertama kali saat aplikasi dijalankan. Pengubahan dilakukan pada baris 1 yaitu saat aplikasi dijalankan akan memanggil kelas *controller* *user_con* yang mengarah pada halaman *log in*. Pada halaman *log in* terdapat dua *text field* dan satu tombol. *Text field* pertama adalah tempat *user* untuk memasukkan *user id* *user*. *Text field* kedua adalah tempat *user* memasukkan *password*, dan tombol *log in* yang digunakan untuk memproses inputan dari *user*. *Username* dan *password* kemudian diproses untuk dilakukan pengecekan ke *database*. Apabila *username* dan *password*

tidak terdapat dalam *database*, akan kembali ke halaman *log in*. Jika cocok akan maka *user* berhasil *log in*. Gambar 7 adalah respons *server* ketika ada *user* yang *log in*. Gambar 8 merupakan halaman yang akan muncul ketika *user* berhasil *log in*.

Gambar 8 menunjukkan halaman yang ditampilkan ketika *user* berhasil *log in*. Pada bagian *header* sebelah kanan terdapat menu untuk *user*. *User* dibagi menjadi 2 hak akses dengan menu yang berbeda. Hak akses yang pertama adalah administrator dan hak akses yang kedua adalah *user* biasa. Perbedaan menu dapat dilihat pada Gambar 9.

Gambar 9 menunjukkan perbedaan hak akses pada tiap-tiap *user*. *User* yang mendapat hak akses sebagai administrator terdapat label “administrator” sedangkan untuk *user* yang tidak mendapat hak akses administrator tidak terdapat label “administrator”. *User* yang mendapat hak administrator terdapat menu panel administrator, pengumuman, *settings*, dan *log out*. Sedangkan *user* biasa hanya terdapat menu *settings* dan *log out*. Kedua hak akses tersebut bisa melakukan *chatting* dengan sesama *user*. Pada bagian di bawah *header* dibagi lagi menjadi 3 bagian yaitu bagian sebelah kiri untuk menampilkan *user* yang *online* maupun yang sedang *offline*. Bagian tengah untuk melakukan *chatting*, dan bagian kanan untuk menampilkan pengumuman. Untuk menampilkan *user* yang sedang *online* maupun *offline* aplikasi akan *mereload* secara berkala halaman *userlist*.

Kode Program 5 menunjukkan fungsi untuk menampilkan *user* yang *online* maupun *offline* dengan memanfaatkan JQuery Ajax, setiap 10000 *millisecond class controller* *userList* akan *direload* secara terus menerus tanpa *merefresh* keseluruhan halaman. Proses *chatting* pada penelitian ini memanfaatkan *library* Javascript Socket.IO. Penggunaan Socket.IO memungkinkan komunikasi secara *realtime* antara *client* dan *server*. Socket.IO berjalan pada *web browser* dan berjalan pada *server*. Kode Program 6 merupakan fungsi pada sisi *client* untuk mengirimkan data dari *client* ke *server*.

Kode Program 6 menunjukkan fungsi untuk mengirimkan data dari *client* ke *server*. Data yang diinputkan akan menjadi parameter pada fungsi ‘*send message*’ pada komputer *client* dan diterima oleh *server*.

Gambar 10 menunjukkan aktivitas *server*. Pesan yang masuk akan ditampilkan ke dalam *console*. *Server* mengirim nilai pengembalian data pada *client* dalam fungsi ‘*whisper*’ dan “*kirim*”. Gambar 11 merupakan hasil data yang ditampilkan di *web browser*. Gambar 12 merupakan notifikasi ketika ada pesan yang belum dibaca.

Gambar 12 menunjukkan notifikasi pesan yang belum dibaca. Ketika ada pesan yang baru saja masuk atau pesan yang belum dibaca, pada *header* akan muncul notifikasi yang berisi jumlah pesan dan siapa pengirim pesan tersebut. *User* dengan hak akses administrator juga bisa melakukan *posting* pengumuman. *Posting* pengumuman juga dilakukan melalui Socket.

Gambar 13 menunjukkan *form* untuk menulis pengumuman. Data inputan akan dikirimkan ke *server*. Kemudian *server* mengolah data dan mengirimkan kembali ke *client* untuk ditampilkan yang terlihat pada Gambar 14.

Aplikasi *chatting* yang dibangun dapat menyesuaikan dengan resolusi layar ketika aplikasi diakses dengan memanfaatkan *Framework* Foundation. Pemanfaatan *Framework* Foundation aplikasi *web* akan menjadi *responsive*.

Grid dalam *framework* Foundation digunakan untuk mengaktifkan fitur *responsive*. Maksimum penggunaan grid pada *Framework* Foundation adalah 12 kolom. Grid dikelompokkan menjadi 3 properti yaitu *small*, *medium*, *large*. Properti *small* digunakan untuk menyesuaikan resolusi layar yang kecil, properti *medium* digunakan untuk menyesuaikan resolusi layar yang lebih besar dari properti *small* dan properti *large* untuk resolusi layar yang lebih besar lagi. Kode Program 7 menunjukkan pengaturan media *query* yang digunakan untuk menampilkan *layout* yang berbeda berdasarkan media yang digunakan.

Gambar 15 menunjukkan tampilan aplikasi ketika diakses pada resolusi layar yang lebih kecil. Dilakukan penambahan menu pada sisi bawah layar untuk mengakses menu melihat daftar *user* dan melihat pengumuman.

Pengujian sistem dilakukan untuk mencari kesalahan pada aplikasi *chatting* yang dibuat. Pengujian dilakukan untuk mengetahui apakah aplikasi yang sudah dibuat berjalan dengan baik dan sesuai dengan kebutuhan *user*. Pengujian alpha menggunakan metode *blackbox* yaitu pengujian fungsi-fungsi aplikasi secara langsung tanpa memperhatikan alur eksekusi program. Pengujian ini dilakukan dengan memeriksa setiap fitur aplikasi. Berdasarkan pengujian yang dilakukan pada aplikasi *chatting*, setiap fungsi dapat berjalan dengan baik, termasuk dapat mengirimkan dan menerima *file attachment* dengan ukuran maksimum 25 MB. Berdasarkan hal ini, maka disimpulkan bahwa aplikasi ini berfungsi dengan baik dan sesuai yang diharapkan.

Pengujian lainnya juga dilakukan untuk menguji performa aplikasi *chatting* ini ketika digunakan oleh banyak *user*. Pengujian dilakukan dengan bantuan aplikasi Web Application Testing (WAPT), yaitu melibatkan 110 *user*. Hasil pengujian ditunjukkan oleh Gambar 16.

Berdasarkan grafik pada Gambar 16, dapat dijelaskan tentang *performance* aplikasi yang meliputi waktu *response* yang dibutuhkan oleh seorang *user* untuk menunggu *response* dari *server*, waktu *download* yang dibutuhkan oleh *user* dan banyaknya halaman *web* yang dapat dieksekusi tiap detik. Hasil pada grafik menunjukkan bahwa waktu *response* aplikasi cenderung tidak stabil. Karena pada interval waktu tertentu terjadi waktu *response* yang singkat yaitu 0 detik dan kadang terjadi waktu *response* yang cenderung lama yaitu 3.03 detik. Dalam analisis waktu *response* terdapat 3 (tiga) batasan, yaitu waktu *response* 0.1 detik merupakan batas ideal bagi *user* dalam menunggu *response* dari *server*, 1.0 detik merupakan batas yang masih bisa ditoleransi dan untuk batas 10 detik merupakan batas yang tidak dapat ditoleransi lagi, karena menurut *survey* waktu *response* yang memasuki detik ke 8.0 akan membuat *user* meninggalkan sistem. Dari grafik dapat disimpulkan bahwa waktu *response* masih dapat ditoleransi, karena berada dalam batas 0.0 - 3.0 detik.

Download time yang dimaksud di sini adalah waktu yang dibutuhkan pada saat *user* melihat *web title* (pada *browser title bar*) hingga *user* dapat melihat halaman *web* yang meliputi *frame*, *tabel* dan *HTML text*. Dari grafik pada Gambar 16 terlihat bahwa *download time* dapat dikatakan stabil (waktu *download* tidak berubah untuk tiap periode waktu) dan singkat (waktu *download* hanya 0 detik).

Pages per second yang dimaksud adalah banyaknya halaman *web* yang dapat dieksekusi tiap detik. Hal ini

terlihat pada grafik pada Gambar 16 di mana untuk tiap interval waktu tertentu tidak terjadi perubahan *pages per second* yang besar. Didapatkan 0.1 halaman tiap detik (paling sedikit) dan 0.3 halaman tiap detik (paling banyak).

Pengujian beta dilakukan dengan melakukan presentasi dengan EDP Keuangan. Presentasi dilakukan dengan melakukan demo program yang telah di-*upload* ke *server* untuk mengetahui apakah aplikasi *chatting* berfungsi dengan baik dan sesuai kebutuhan pengguna. Berdasarkan hasil presentasi dengan karyawan EDP Keuangan, aplikasi berfungsi dengan baik dan sesuai dengan kebutuhan pengguna.

5. Kesimpulan

Berdasarkan penelitian yang telah dilakukan maka dapat diambil kesimpulan bahwa pembuatan aplikasi *chatting* dapat dibuat dengan menggunakan *Framework* CodeIgniter, Socket.IO dan *Framework* Foundation. Pemanfaatan *Framework* CodeIgniter dapat diterapkan konsep MVC (*Model View Controller*) sehingga penulisan kode menjadi lebih terstruktur dan terorganisir. Pemanfaatan Socket.IO cocok untuk pembuatan *aplikasi realtime* seperti aplikasi *chatting* dan *Framework* Foundation membuat tampilan aplikasi menjadi *responsive* yang dapat menyesuaikan diberbagai resolusi layar.

Berdasarkan hasil pengujian, aplikasi *chatting* dapat membantu karyawan EDP Keuangan atau *user* lainnya untuk berkomunikasi, *attach file* dengan ukuran maksimum 25 MB, dan menyebarkan informasi jika sewaktu-waktu terjadi penambahan atau perubahan program yang dikembangkan oleh karyawan EDP Keuangan. Spesifikasi *hardware* maupun *software* aplikasi *chatting* ini tergolong aplikasi yang ringan karena merupakan aplikasi berbasis *web*, *user* hanya membutuhkan *web browser* kemudian mengakses aplikasi *chatting* yang sudah *upload* ke *server* dan dengan memanfaatkan jaringan lokal maka untuk mengakses aplikasi *chatting* tidak membutuhkan koneksi internet.

Saran pengembangan yang dapat dilakukan adalah dengan menambahkan fitur keamanan pada aplikasi *chatting*. Fitur keamanan ini digunakan untuk mengenkripsi pesan *chat* yang dikirimkan. Aplikasi *chatting* juga bisa dikembangkan untuk versi aplikasi *mobile* yang tersinkronisasi ke aplikasi *chatting web* ini, sehingga *user experience* menjadi lebih baik.

6. Daftar Pustaka

- [1] M. Z. Teddy, dan D. W. Surya, "Aplikasi Chat pada Handphone dan Komputer dengan Media Bluetooth (Bluetooth Chat)," Jurnal Teknologi Informasi - Aiti, vol. 4, no. 1, pp. 60-74, 2009.
- [2] S. Roni, dan S. Edhy, "Membangun Aplikasi Chatting Berbasis Multiuser," Jurnal Dasi, vol. 10, no. 1, pp.3-22, 2009.
- [3] E. Sularso, W. S. Raharjo, & Y. Lukito, "Implementasi Algoritma RijnDeal 128 pada Aplikasi Chatting Berbasis HTML WebSocket," Jurnal Infomatika, vol. 10, no. 2, pp. 66-79, 2014.

- [4] Priyanto, D. *Mahir Komputer Tanpa Kursus. Belajar Mudah Internet*. Yogyakarta: MediaKom, 2009.
- [5] Wardana. *Menjadi Master PHP dengan Framework CodeIgniter*. Jakarta: Elex Media Komputindo, 2010.
- [6] Anonymous. Nodsource. [Online], <https://nodsource.com/blog/understanding-socketio/>, tanggal akses 6 Juli 2016.
- [7] Anonymous. Foundation Zurb. [Online], <http://foundation.zurb.com/learn/why-foundation.html/>, tanggal akses 6 Juli 2016.
- [8] Nugroho, A. *Mengembangkan Aplikasi Basis Data Menggunakan C# dan SQL Server*. Yogyakarta: Andi, 2010.
- [9] Hasibuan, Z. A. *Metodologi Penelitian Pada Bidang Ilmu Komputer dan Teknologi Informasi: Konsep, Teknik, dan Aplikasi*. Jakarta: Ilmu Komputer Universitas Indonesia, 2007.
- [10] Pressman, R. S. *Software Engineering: A Practitioner's Approach*, 2001.

Prototype Sistem Informasi Pelayanan Bayi Baru Lahir pada Fasilitas Kesehatan Primer

Rinda Nurul Karimah*, Andri Permana Wicaksono

Jurusan Kesehatan
Politeknik Negeri Jember
Jember
*rindank6@gmail.com

Abstrak-Pelayanan kesehatan untuk bayi baru lahir merupakan salah satu program kesehatan yang bertujuan untuk menjamin kelangsungan hidup dan tumbuh kembang anak secara optimal. Data Badan Pusat Statistik (BPS) pada tahun 2018 berdasarkan Survei Demografi Kesehatan Indonesia (SDKI) tahun 2017 menunjukkan Angka Kematian Bayi (AKB) di Indonesia masih tinggi yaitu mencapai 24 kematian setiap 1000 kelahiran hidup. Fasilitas kesehatan primer merupakan sarana pelayanan kesehatan tingkat pertama yang berperan sebagai tonggak pelayanan awal, yaitu menjadi gerbang pelayanan kesehatan tingkat lanjut terutama di era jaminan kesehatan nasional seperti saat ini. Penelitian ini merupakan penelitian kualitatif dengan metode desain sistem menggunakan model prototipe. Penerapan pedoman pelayanan kesehatan yang ditetapkan oleh Kementerian Kesehatan bagi pelayanan ibu dan bayi baru lahir di fasilitas kesehatan primer, memerlukan dukungan teknologi informasi berupa sistem informasi untuk mendukung kelengkapan data, sehingga informasi yang dihasilkan dapat lengkap. Hal ini disebabkan oleh adanya tuntutan kebutuhan informasi yang semakin meningkat, terutama terkait adanya permasalahan ketidaklengkapan data pada pelayanan klinis dan pelaporan bayi baru lahir. Hasil keluaran formulir bayi baru lahir dan surat keterangan kelahiran yang diinginkan oleh *user* telah terpetakan dan menjadi *output* dari desain prototipe, sehingga hasil desain prototipe dapat menjadi solusi dari permasalahan yang ada. Penelitian ini dilaksanakan selama tiga bulan, diawali dengan identifikasi kebutuhan, pembangunan prototipe dan diakhiri dengan evaluasi. Luaran yang dihasilkan berupa desain prototipe pendukung kelengkapan data pelayanan klinis bayi baru lahir untuk fasilitas kesehatan primer.

Kata Kunci: Pelayanan bayi baru lahir, Fasilitas Kesehatan Primer, Prototipe.

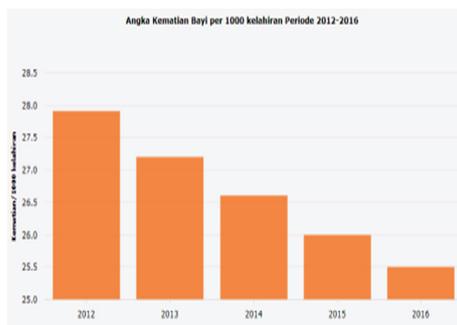
1. Pendahuluan

Bayi baru lahir merupakan kelompok masyarakat yang rentan dan perlu mendapat perhatian serius, karena masih tingginya Angka Kematian Bayi (AKB) di Indonesia. Angka Kematian Bayi Baru Lahir mencapai 2/3 dari total Angka Kematian Bayi [1]. Pelayanan kesehatan untuk bayi baru lahir merupakan salah satu program kesehatan yang bertujuan untuk menjamin kelangsungan hidup dan tumbuh kembang anak secara optimal. Hal ini dilakukan dalam rangka mewujudkan anak Indonesia yang sehat sebagai modal dasar bagi pembangunan bangsa. Hasil riset Badan Pusat Statistik (BPS) di tahun 2016 menunjukkan AKB masih tinggi yaitu mencapai 25,5 artinya ada sekitar 25,5 kematian setiap 1000 bayi lahir [2]. Angka kematian bayi pada tahun 2017 masih tinggi meskipun telah mengalami penurunan menjadi 24 kematian setiap 1000 kelahiran hidup. Data BPS tersebut berdasarkan Survei Demografi Kesehatan Indonesia (SDKI) tahun 2017

Bayi baru lahir memerlukan asuhan yang segera, cepat, tepat, aman dan bersih. Hal tersebut merupakan bagian esensial bayi baru lahir. Sebagian besar proses persalinan terfokus pada ibu, tetapi penatalaksanaan persalinan baru dikatakan berhasil jika ibu dan bayinya dalam kondisi keadaan sehat optimal. Fasilitas kesehatan primer merupakan sarana pelayanan kesehatan tingkat pertama yang berperan sebagai tonggak pelayanan primer,

yaitu menjadi gerbang pelayanan kesehatan tingkat lanjut terutama di era jaminan kesehatan nasional saat ini. Bidan di fasilitas kesehatan primer diberi tanggung jawab penuh terhadap keselamatan ibu dan bayi pada persalinan normal dan beberapa saat sesudah selesainya persalinan. Bidan harus mengetahui dengan segera timbulnya perubahan pada bayi dan bila perlu memberikan pertolongan pertama seperti menghentikan perdarahan, membersihkan jalan napas, memberikan oksigen, dan melakukan pernapasan buatan sampai bayi tersebut mendapat perawatan yang memiliki perlengkapan yang lengkap serta perawatan yang baik, sampai pengawasan dan pengobatan yang dilakukan sebaik-baiknya [3]. Semua kegiatan yang dilakukan memerlukan proses pencatatan terkait kebutuhan kelengkapan data untuk proses dokumentasi medis.

Adanya kejadian penculikan dan perdagangan bayi yang termasuk kriteria perdagangan orang (*trafficking in person*) juga harus menjadi perhatian khusus dan menambah urgensi kebutuhan kelengkapan data pada pelayanan bayi baru lahir. Kejadian ini merupakan suatu tantangan bagi upaya peningkatan kualitas program kesehatan bayi baru lahir yang saat ini menjadi prioritas utama pembangunan kesehatan. Peningkatan jaminan keamanan, prosedur, dan kelengkapan data pada pelayanan bayi baru lahir diharapkan dapat mencapai target pemerintah untuk tidak terjadi lagi kasus penculikan dan perdagangan bayi baru lahir (*zero toleran*) di fasilitas pelayanan kesehatan [1].



Gambar 1. Angka Kematian Bayi di Indonesia Periode 2012-2016

Perkembangan teknologi informasi saat ini sangat terasa manfaatnya dalam membantu permasalahan pada suatu proses kegiatan, tidak terkecuali di bidang pelayanan kesehatan. Hal ini disebabkan oleh adanya tuntutan kebutuhan informasi yang semakin meningkat, terutama terkait kelengkapan data untuk keperluan medis. Teknologi komputerisasi mempunyai kemampuan sebagai media komunikasi yang dapat mempercepat proses kerja manusia, mulai dari pencatatan data, pengolahan data sampai menjadi informasi dan pelaporan yang lengkap dan akurat.

Kementerian Kesehatan telah menetapkan berbagai Peraturan Menteri Kesehatan dan menyusun Pedoman Pelayanan Kesehatan bagi Ibu dan Bayi Baru Lahir di fasilitas kesehatan primer dan bidan jejaring. Pedoman tersebut dipergunakan sebagai acuan bagi tenaga kesehatan dalam memberikan pelayanan, salah satunya terkait Pedoman Asuhan Bayi Baru Lahir yang mengatur standar pelayanan yang bersifat teknis medis. Langkah-langkah kegiatannya berupa: 1) Pendataan bayi baru lahir; 2) Pelayanan kesehatan bayi baru lahir; 3) Pengisian dan pemanfaatan Buku KIA; 4) Pencatatan dan pelaporan; 5) Rujukan pertolongan kasus komplikasi pada bayi baru lahir jika diperlukan [4]. Penerapan standar tersebut memerlukan dukungan teknologi informasi untuk mendukung kelengkapan data, sehingga informasi yang dihasilkan dapat akurat. Keakuratan informasi akan berdampak pada peningkatan keamanan dan kualitas pelayanan medis serta fungsi medikolegal.

Berdasarkan latar belakang tersebut, peneliti mengusulkan solusi permasalahan kebutuhan kelengkapan data dengan membuat perancangan *prototipe* pendukung kelengkapan data pelayanan klinis bayi baru lahir untuk fasilitas kesehatan primer.

2. Metode

a. Jenis Penelitian

Jenis penelitian ini adalah kualitatif dengan model perancangan sistem menggunakan *prototyping model*, yaitu model perancangan yang cepat dan pengujian terhadap model kerja (*prototipe*) dari aplikasi baru melalui proses interaksi dan berulang-ulang yang biasa digunakan ahli sistem informasi dan ahli bisnis [5]. Dengan metode *prototyping* ini pengembang dan pengguna (*user*) dapat saling berinteraksi selama proses pembuatan sistem [3]. Model *prototipe* pada penelitian ini terdiri dari langkah sebagai berikut:

1. Pengumpulan kebutuhan/identifikasi kebutuhan
2. Membangun *prototyping*
3. Evaluasi *prototyping*

b. Lokasi Penelitian

Penelitian ini dilakukan selama 3 (tiga) bulan yaitu dari bulan Agustus sampai Oktober 2017 di Klinik Sakinah Kaliurang, yaitu salah satu FKTP di Kabupaten Jember yang memiliki keanggotaan kapitasi peserta BPJS mencapai 600 orang dan memiliki 2 orang bidan satu atap dan 3 bidan jejaring yang tersebar di wilayah Kecamatan Sumber Sari.

c. Unit Analisis

Unit analisis pada penelitian ini adalah seluruh bidan yang bertugas melaksanakan pelayanan bayi baru lahir di FKTP klinik Sakinah Kaliurang Jember. Terdapat lima orang bidan (dua orang bidan satu atap dan tiga bidan jejaring yang tersebar di wilayah Kecamatan Sumber Sari).

d. Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan dalam penelitian ini meliputi observasi, wawancara, dan diskusi. Analisis data dilakukan secara kualitatif dengan mengeksplorasi pengetahuan para bidan yang diperoleh melalui wawancara dan diklarifikasi dengan hasil observasi. Hasil tersebut kemudian dilakukan diskusi dan dikombinasikan dengan referensi terkait regulasi yang berlaku dari Kemenkes RI untuk memperoleh kesimpulan terkait kebutuhan pembangunan prototipe.

3. Hasil

Hasil dari penelitian ini akan dilakukan tiga langkah model desain sistem prototipe, yaitu:

1. Tahap pengumpulan kebutuhan/identifikasi kebutuhan diperoleh hasil kebutuhan responden terkait dukungan saat pelaksanaan pelayanan klinis bayi baru lahir berupa kewaspadaan umum (*universal precaution*) saat penilaian awal sebelum bayi lahir dan segera setelah bayi baru lahir. Dukungan yang dibutuhkan berupa alur tindakan terorganisir sebagai implementasi MTBS (Manajemen Terpadu Balita Sakit) dan kebutuhan keseragaman formulir terkait instrumen penangkapan kelengkapan data serta terkait legalitas dari pendokumentasian yang dilakukan termasuk pengeluaran surat keterangan kelahiran.
2. Tahap membangun/pengembangan *prototyping*, pada tahap ini dihasilkan *Data Flow Diagram* (DFD) dan *Entity Relation Diagram* (ERD) Prototipe pendukung kelengkapan data pelayanan klinis bayi baru lahir di fasilitas kesehatan primer.
3. Evaluasi *prototyping*, hasil evaluasi yang diperoleh adalah rancangan DFD dan ERD pada tahap sebelumnya dapat diterima oleh *user* kemudian disajikan dalam *interface* sistem.

4. Diskusi

a. Identifikasi Kebutuhan

Identifikasi kebutuhan yang dimaksud adalah kebutuhan terhadap prototipe pendukung kelengkapan

data pelayanan klinis bayi baru lahir untuk fasilitas kesehatan primer. Pengambilan data melalui kegiatan observasi, wawancara, dan diskusi dengan klinisi pelaksana pelayanan klinis bayi baru lahir di fasilitas kesehatan primer dalam hal ini khususnya klinisi di klinik Sakinah Kaliurang dan bidan jejaringnya. Terdapat 5 orang responden pada penelitian ini yaitu 2 orang bidan satu atap dan 3 bidan jejaring.

Hasil kesimpulan wawancara yang diperoleh dikombinasikan dengan referensi terkait regulasi yang berlaku dari Depkes yaitu Peraturan Menteri Kesehatan Republik Indonesia Nomor 53 Tahun 2014 Tentang Pelayanan Kesehatan Neonatal Esensial pasal 4 ayat 2 tentang pelayanan neonatal esensial 0 (no) sampai 6 (enam) jam meliputi:

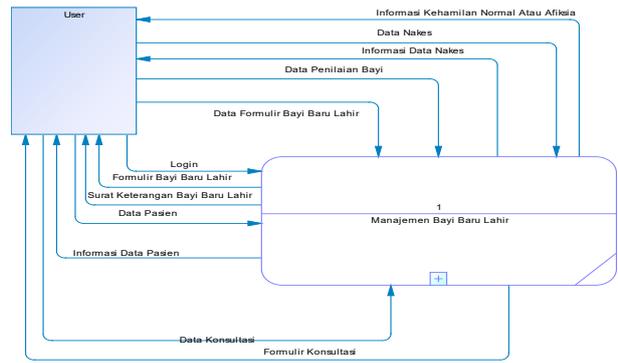
- 1) Menjaga bayi tetap hangat;
- 2) Inisiasi menyusui dini;
- 3) Pemotongan dan perawatan tali pusat;
- 4) Pemberian suntikan vitamin k1;
- 5) Pemberian salep mata antibiotik;
- 6) Pemberian imunisasi hepatitis b0;
- 7) Pemeriksaan fisik bayi baru lahir;
- 8) Pemantauan tanda bahaya;
- 9) Penanganan asfiksia bayi baru lahir;
- 10) Pemberian tanda identitas diri;
- 11) Merujuk kasus yang tidak dapat ditangani dalam kondisi stabil, tepat waktu ke fasilitas pelayanan kesehatan yang lebih mampu [7].

Hal tersebut juga sesuai dengan regulasi yang terdapat pada panduan pelayanan kesehatan neonatal esensial, sehingga disimpulkan kebutuhan data yang diperlukan terkait pengembangan *prototipe* pendukung kelengkapan data pelayanan klinis bayi baru lahir untuk fasilitas kesehatan primer berupa bagan alur manajemen bayi baru lahir yang terbagi menjadi bayi baru lahir normal dan bayi baru lahir dengan asfiksia. Asuhan bayi baru lahir dan tindakan penilaiannya. Selain itu juga formulir panduan pemeriksaan fisik yang dilakukan pada bayi baru lahir, formulir bayi baru lahir, dan penerbitan surat keterangan lahir.

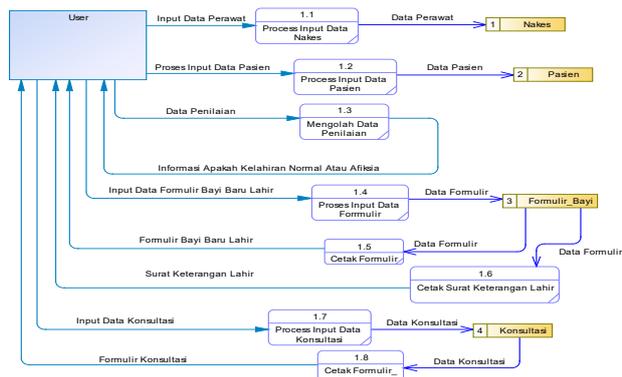
Hasil keluaran formulir bayi baru lahir dan surat keterangan kelahiran yang diinginkan oleh *user* telah terpetakan dan menjadi *output* dari desain prototipe. Formulir tersebut berisi data-data pelayanan secara lengkap dan terstruktur disertai otentifikasi legalitas terdiri dari pemberi layanan, penerima layanan, dan saksi.

b. Mengembangkan Prototipe

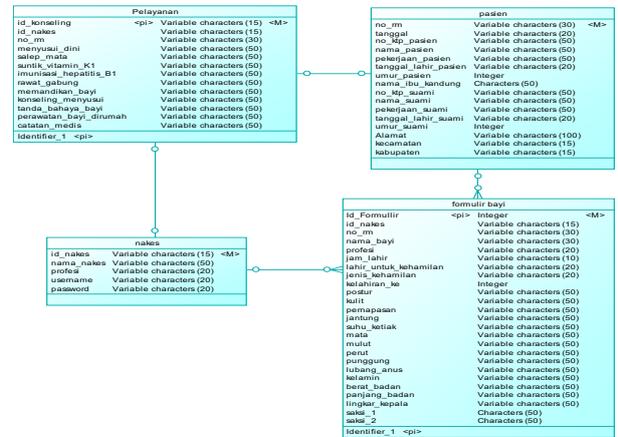
Pengembangan prototipe dilakukan dengan *mapping* kebutuhan data terkait pelayanan klinis bayi baru lahir yang telah diperoleh pada tahap sebelumnya (tahap identifikasi kebutuhan), kemudian membuat mekanisme inferensi untuk mendeteksi kebutuhan pelayanan. Sistem pendukung operasional yang dihasilkan nantinya dapat menghasilkan bermacam produk informasi baik untuk pengguna internal maupun pengguna eksternal [8]. Pengguna internal pada prototipe yang dikembangkan ini adalah tenaga kesehatan terkait pelayanan bayi baru lahir, sedangkan pengguna eksternal merupakan pasien, keluarga pasien dan pihak ketiga. Tahap ini dihasilkan *Data Flow Diagram* (DFD) (Gambar 2)



Gambar 2. DFD Level 0 Prototipe Pendukung Kelengkapan Data Pelayanan Klinis Bayi Baru Lahir



Gambar 3. DFD Level 1 Prototipe Pendukung Kelengkapan Data Pelayanan Klinis Bayi Baru Lahir



Gambar 4. ERD Prototipe Pendukung Kelengkapan Data Pelayanan Klinis Bayi Baru Lahir

DATA PASIEN

No RM: Tanggal:

No KTP Pasien: <input type="text"/>	No KTP Suami: <input type="text"/>	Tambah	Simpan
Nama Pasien: <input type="text"/>	Nama Suami: <input type="text"/>	Udah	Batal
Pekerjaan Pasien: <input type="text"/>	Pekerjaan Suami: <input type="text"/>	Keluar	Riplus
Umur Pasien: <input type="text"/>	Umur Suami: <input type="text"/>		
Alamat: <input type="text"/>	Nama Ibu Kandung: <input type="text"/>		
	Kecamatan: <input type="text"/>		
	Kabupaten / Kota: <input type="text"/>		

Gambar 5. Interface Data Pasien

DATA NAKES

Id Nakes : <input type="text"/>	<input type="button" value="Tambah"/> <input type="button" value="Simpan"/> <input type="button" value="Ubah"/> <input type="button" value="Batal"/> <input type="button" value="Keluar"/> <input type="button" value="Hapus"/>
Nama Nakes : <input type="text"/>	
Jenis Kelamin : <input type="text"/>	
Profesi : <input type="text"/>	
Username : <input type="text"/>	
Password : <input type="text"/>	

Gambar 6. *Interface* Data Nakes

Formulir Bayi Baru Lahir

Id Formulir : <input type="text"/> Id Perawat : <input type="text"/> No RM : <input type="text"/> Nama Bayi : <input type="text"/> Jenis Kelamin Bayi : <input type="radio"/> Laki - Laki <input type="radio"/> Perempuan Tanggal Lahir Bayi : <input type="text"/>	Lahir Pada Umur Kehamilan : <input type="text"/> Jenis Kelahiran : <input type="radio"/> Tunggal <input type="radio"/> Kembar 2 <input type="radio"/> Kembar 3 <input type="radio"/> Lainnya Kelahiran Ke : <input type="text"/> Jam Lahir Bayi : <input type="text"/>
1. Postur Tonus & Aktivitas : <input type="text"/> 9. Pent dan Tali Pusat : <input type="text"/> 2. Kulit Bayi : <input type="text"/> 10. Punggung Tulang Belakang : <input type="text"/> 3. Napas Jika Tidak Nangis : <input type="text"/> 11. Lubang Anus : <input type="text"/> 4. Detak Jantung : <input type="text"/> 12. Alat Kelamin : <input type="text"/> 5. Suhu Ketiak : <input type="text"/> 13. Berat Badan : <input type="text"/> 6. Kepala : <input type="text"/> 14. Panjang Badan : <input type="text"/> 7. Mata : <input type="text"/> 15. Lingkar Kepala : <input type="text"/> 8. Mulut (Lidah, Selaput Lendir) : <input type="text"/>	
Saksi I <input type="text"/> Saksi 2 <input type="text"/>	

Gambar 7. *Interface* Inputan Formulir Bayi Baru Lahir

FORMULIR KONSULTASI BAYI

Id Konseling : <input type="text"/>	No RM : <input type="text"/>	Id Nakes : <input type="text"/>
1. Inisiasi Menyusui Dini? <input type="radio"/> Iya <input type="radio"/> Tidak 2. Salep Mata Antibiotika Profilaksis? <input type="radio"/> Iya <input type="radio"/> Tidak 3. Suntikan Vitamin K1? <input type="radio"/> Iya <input type="radio"/> Tidak 4. Imunisasi Hepatitis B1? <input type="radio"/> Iya <input type="radio"/> Tidak 5. Rawat Gabung Dengan Ibu? <input type="radio"/> Iya <input type="radio"/> Tidak 6. Memandikan Bayi? <input type="radio"/> Iya <input type="radio"/> Tidak 7. Konseling Menyusui? <input type="radio"/> Iya <input type="radio"/> Tidak 8. Tanda Bahaya Pada Bayi Yang Perlu Dirujuk? <input type="radio"/> Iya <input type="radio"/> Tidak 9. Menjelaskan Pada Ibu Tentang Perawatan Bayi Di Rumah? <input type="radio"/> Iya <input type="radio"/> Tidak 10. Melengkapi Catatan Medis? <input type="radio"/> Iya <input type="radio"/> Tidak		
<input type="button" value="CARI DATA NAKES"/> <input type="button" value="CARI DATA PASIEN"/> <input type="button" value="TAMBAH"/> <input type="button" value="SIMPAN"/> <input type="button" value="BATAL"/>		

Gambar 8. *Interface* Inputan Formulir Konsultasi Bayi

PENILAIAN BAYI LAHIR NORMAL ATAU ASFIKZIA

1. Apakah Bayi Bernapas Dengan Normal / Tidak Megap - Megap? Iya Tidak

Ventilasi

2. Apakah Bayi Bernapas Dengan Normal / Tidak Megap - Megap? Iya Tidak

Asuhan Pasca Resusitasi

3. Apakah Bayi Ingin Dirujuk / Tidak Berhasil? Iya Tidak

Rujukan Bayi

Lanjut

Gambar 9. *Interface* Inputan Penilaian bayi Form 1

Diagram alir data adalah teknik grafik yang digunakan untuk menjelaskan aliran informasi dan transformasi data dari masukan hingga keluaran [4]. DFD level 0 atau biasa juga disebut diagram konteks merupakan gambaran bagaimana sistem berinteraksi dengan eksternal *entity*, dalam hal ini yaitu *user* yang terdiri dari bidan pelaksana kegiatan terkait pelayanan klinis bayi baru lahir di fasilitas kesehatan primer.

1. Apakah Kehamilan Cukup Bulan ?	<input checked="" type="radio"/> Iya <input type="radio"/> Tidak
2. Apakah Air Ketuban Jernih Tidak Kecampur Mekonium ?	<input checked="" type="radio"/> Iya <input type="radio"/> Tidak
3. Apakah Bayi Menangis Bemapas / Tidak Megap - Megap ?	<input checked="" type="radio"/> Iya <input type="radio"/> Tidak
4. Apakah Tonus Otot Bayi Baik / Bergerak Aktif ?	<input checked="" type="radio"/> Iya <input type="radio"/> Tidak

KETERANGAN BAYI LAHIR DENGAN NORMAL

Gambar 10. *Interface* Inputan Penilaian bayi Form 2 dan Keterangan Hasil

FORMULIR BAYI BARU LAHIR

Pemeriksaan (Nama dan Tanda Tangan) : _____
 Nama Bayi : _____ Jenis Kelamin : _____
 Nama Orang Tua : _____ Alamat : _____
 NIK Orang Tua Kandung : _____ No RM : _____
 Tanggal & Jam Lahir : _____ Lahir Pada Umur Kehamilan : _____

Pemeriksaan	Tanggal _____ Jam _____	Tanggal _____ Jam _____
	Hasil	Hasil
1. Postur, tonus, dan aktivitas		
2. Kulit bayi		
3. Pemapasan ketika bayi sedang tidak menangis		
4. Derak jantung		
5. Suhu ketiak		
6. Kepala		
7. Mata		
8. Mulut (lidah, selaput lendir)		
9. Perut dan tali pusat		
10. Punggung tulang belakang		
11. Lubang anus		
12. Alat kelamin		
13. Berat badan		
14. Panjang badan		
15. Lingkar kepala		
ASUHAN / KONSELING	Waktu (tanggal, jam) dilakukan asuhan	Keterangan
1. Inisiasi menyusui dini		
2. Salep mata antibiotika profilaksis		
3. Suntukan vitamin K1		
4. Imunisasi hepatitis B1		
5. Rawat gabung dengan ibu		
6. Memandikan bayi		
7. Konseling menyusui		
8. Tanda - tanda bahaya pada bayi yang perlu dirujuk		
9. Menjelaskan pada ibu tentang perawatan bayi di rumah		
10. Melengkapi catatan medis		
Waktu pemeriksaan kembali / kunjungan neonatal	Tanggal _____	Tanggal _____

Nama Orang Tua Kandung : _____ Saksi : _____ Penolong Persalinan : _____

(.....) (.....) (.....)

No : _____

Yang bertanda tangan di bawah ini, menerangkan bahwa :
 Pada hari ini _____, Tanggal _____, Pukul _____
 Telah lahir seorang bayi:

Jenis Kelamin : _____
 Jenis Kelahiran : _____
 Kelahiran Ke : _____
 Berat Lahir : _____
 Panjang Badan : _____
 Di Rumah Sakit / Puskesmas / Rumah Bersalin / Polindes / Rumah Bidan / di* _____

Alamat : _____
 Diben Nama : _____

Dari Orang Tua : _____
 Nama Ibu : _____
 Pekerjaan : _____
 KTP / NIK No. : _____
 Nama Ayah : _____
 Pekerjaan : _____
 Pekerjaan : _____
 KTP / NIK No. : _____
 Alamat : _____
 Kecamatan : _____
 Kab. / Kota : _____

Saksi I _____, Tanggal _____
 Saksi II _____
 Penolong Persalinan _____

Gambar 11. *Output* Formulir dan Surat Keterangan Lahir

DFD level 1 (gambar 3) menunjukkan penjabaran lebih rinci dari DFD level 0 terakut masukan-proses dan keluaran pada sistem pendukung kelengkapan data pelayanan klinis bayi baru lahir. DFD dalam bahasa Indonesia disebut sebagai DAD (Data Arus Data) memperlihatkan gambaran tentang masukan-proses-keluaran dari suatu sistem/perangkat lunak, kemudian ditransformasi oleh elemen-elemen pemrosesan dan objek-objek data mengalir ke dalam perangkat lunak, kemudian ditransformasi oleh elemen-elemen pemrosesan, dan objek-objek data hasilnya akan mengalir keluar dari sistem atau perangkat lunak [6].

Gambar 4 menunjukkan desain *Entity Relation Diagram* (ERD) Prototipe Pendukung Kelengkapan Data Pelayanan Klinis Bayi Baru Lahir di FKTP. ERD adalah suatu model jaringan kerja (*network*) yang menguraikan susunan data yang disimpan dari sistem secara abstrak [9]. Gambar ERD berikut menjelaskan hubungan antardata dalam basis data yang mempunyai hubungan antarrelasi yaitu antara entitas nakes (tenaga kesehatan) dalam hal ini bidan atau dokter penanggung jawab pelaksana pelayanan dengan entitas pasien, pelayanan, dan formulir bayi.

c. Evaluasi Prototipe

Hasil dari pengembangan prototipe berupa DFD dan ERD dievaluasi untuk mengetahui apakah dapat diterima oleh *user*. *User* pada penelitian ini adalah responden yang akan menggunakan aplikasi hasil operasional dari prototipe yang dikembangkan. Evaluasi dilakukan melalui kegiatan diskusi kembali dengan para responden yaitu bidan pelaksana pelayanan klinis bayi baru lahir. Hasil evaluasi yang diperoleh adalah rancangan DFD dan ERD dapat diterima oleh responden. DFD dan ERD yang telah diterima tersebut kemudian dilanjutkan dengan penyajian dalam bentuk gambaran desain *interface* (tampilan antar muka) sistem (Gambar 5)

Interface data pasien ditunjukkan pada gambar 5 yang berisi data sosial pasien dan pemberian nomor rekam medis serta tanggal pelayanan.

Interface data tenaga kesehatan ditunjukkan pada gambar 6, di mana untuk tenaga kesehatan yang dimaksud adalah terdiri dari menu isian profesi bidan atau dokter yang memberi pelayanan.

Interface inputan data formulir bayi baru lahir ditunjukkan pada gambar 7, di mana berisi informasi detail terkait data sosial dan kondisi klinis bayi berdasarkan hasil pemeriksaan secara objektif. Sedangkan *interface* inputan formulir konsultasi bayi disajikan pada gambar 8, di mana disediakan menu opsi pilihan ya dan tidak sesuai dengan pelayanan yang telah diberikan. Pada *interface* ini juga disediakan menu pencarian data baik data nakes maupun data pasien, hal ini ditujukan untuk memudahkan dengan pencarian kembali dari *data base* hasil *input* pada gambar 5 dan 6.

Interface inputan formulir penilaian bayi disajikan pada gambar 9, di mana pada *interface* ini *user* dituntun dengan arahan tindakan sesuai prosedur terkait dengan pilihan kondisi klinis dari bayi baru lahir.

Lanjutan dari arahan tindakan berupa item pertanyaan terkait penilaian bayi dengan hasil akhir menu berupa tampilan keterangan bayi, apakah bayi lahir dengan kondisi

normal atau disertai asfiksia. Contoh tampilan penilaian dengan hasil akhir keterangan bayi lahir dengan normal nampak pada gambar 10 Selain itu *output* berupa cetak formulir terdiri dari formulir bayi baru lahir dan formulir surat keterangan lahir (Gambar 11).

5. Kesimpulan

Hasil penelitian dan pembahasan yang diperoleh dapat disimpulkan yaitu hasil perancangan prototipe secara keseluruhan dapat dikatakan berhasil, karena telah dapat diterima oleh *user*. Hal ini sesuai dengan kebutuhan dari nakes pemberi pelayanan pada bayi baru lahir di fasilitas kesehatan primer terkait permasalahan kebutuhan kelengkapan data untuk pelayanan klinis bayi baru lahir di fasilitas kesehatan primer. Saran dari peneliti, diharapkan hasil rancangan prototipe ini dapat dilanjutkan ke tahap selanjutnya yaitu digunakan dalam operasional sehingga dihasilkan aplikasi kelengkapan data untuk pelayanan klinis bayi baru lahir di fasilitas kesehatan primer.

6. Daftar Pustaka

- [1] Direktorat Kesehatan Anak Khusus, Panduan Pelayanan Kesehatan Bayi Baru Lahir Berbasis Perlindungan Anak. KEMENKES, Jakarta, 2010,
- [2] Badan Pusat Statistik, Angka Kematian Bayi, BPS Jakarta, 2018.
- [3] Djauhari T, Assegaff S, 2016. Perancangan Sistem Informasi Layanan Persidangan Pada Pengadilan Negeri Sengeti. Jurnal Manajemen Sistem Informasi Vol. 1 No. 1, September 2016.
- [4] Mulyanto agus, 2009. Sistem Informasi Konsep dan Aplikasi. Pustaka Pelajar.
- [5] Setiawan, Prototipe Sistem Informasi Inventori Dengan Pendekatan Berorientasi Objek, Jurnal Letera ICT, Vol.3 No.1, Mei 2016
- [6] Nurhasanah Mey, dkk, Asuhan Kebidanan Pada Bayi Baru Lahir Fisiologis Di BPM Bidan Cucu Hudami, Amd.Keb Kabupaten Ciamis, Jurnal STIKES CIAMIS, 2015. Available from: http://www.ejournal.stikesmucis.ac.id/file.php?file=preview_mahasiswa&cid=1067&c-d=0b2173ff6ad6a6fb09c95f6d50001df6&name=13DB277070.pdf
- [7] KEMENKES RI, Buku Kesehatan Ibu dan Anak. Kementerian Kesehatan RI dan JICA. Jakarta, 2016.
- [8] O'Brien, A. James, Marakas, George.M, Management Information System, 10th Edition McGraw-Hill/Irwin, New York, 2010.
- [9] Yuhendra, Eko Yulianto R, 2015. Rekayasa Perangkat Lunak Pengolahan Data Distribusi Obat-obatan di PT. Anugrah Pharmindo Lestari Berbasis WEB. Jurnal Momentum, Vol. 17 No. 2, Agustus 2015.

Rekomendasi Paket Produk Guna Meningkatkan Penjualan Dengan Metode FP-Growth

Asrul Abdullah*

Teknik Informatika

Universitas Muhammadiyah Pontianak

Pontianak

*asrul.abdullah@unmuhpnk.ac.id

Abstrak-Setiap pedagang atau pemilik tempat usaha harus dapat menemukan strategi agar tempat usahanya mampu bertahan dan terus berkembang. Keinginan konsumen dalam membeli suatu produk memang sangat beragam. Ada pembeli yang gemar membeli produk yang telah dipaket, tetapi ada juga pembeli yang membeli produk yang mendapat diskon, dan masih banyak lagi. Keragaman perilaku konsumen dalam memilih barang yang ingin dibeli tentunya menjadi permasalahan tersendiri bagi para penjual jika ingin meningkatkan penjualannya. Algoritma FP-Growth sendiri merupakan hasil perbaikan dari algoritma Apriori. FP-Growth membangkitkan struktur data *tree* atau yang lebih dikenal sebagai *frequent pattern tree*. FP *tree* memerlukan dua kali *scanning database* untuk menentukan *frequent itemset* sehingga membuatnya lebih efisien dibandingkan Apriori. Hasil dari *association rules* menggunakan FP-Growth dijadikan rekomendasi bagi para penjual/*retailer* dalam memberikan paket penjualan barang bagi konsumennya dan hasil dari penelitian ini adalah ditemukan dua pasangan item barang yakni kopi, gula dan teh, susu yang memiliki *support* sebesar 30% dan *confidence* sebesar 70%.

Kata Kunci: FP-Growth; frequent itemset; association rules

1. Pendahuluan

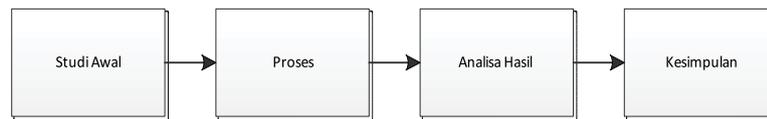
Perkembangan pasar maya (*online*) dan pasar tradisional membuat persaingan dalam menarik konsumen menjadi semakin ketat. Setiap pedagang atau pemilik usaha harus dapat menemukan strategi agar tempat usahanya mampu bertahan dan terus berkembang. Salah satu strategi yang bisa diterapkan adalah pedagang harus dapat mengetahui keinginan konsumen atau pedagang dapat memberikan pilihan produk yang sesuai dengan kebutuhan konsumen [1]. Kemampuan pedagang dalam melihat dua hal tersebut yakni keinginan dan kebutuhan konsumen pastinya berimbas pada peningkatan penjualan dari sebuah produk. Keinginan dan kebutuhan konsumen dalam membeli suatu produk memang sangat beragam. Sebagai contoh, saat kita membeli produk di toko *retail*, pastinya calon pembeli ditawarkan beragam produk mulai dari produk paketan, produk pelengkap, dan produk yang mendapat diskon. Berbagai cara menawarkan produk akan memicu calon pembeli untuk membeli produk tersebut. Ada pembeli yang gemar membeli produk yang telah dipaket, tetapi ada juga pembeli yang membeli produk yang mendapat diskon, dan masih banyak lagi. Keragaman keinginan konsumen dalam memilih barang yang ingin dibeli tentunya menjadi permasalahan tersendiri bagi para penjual jika ingin meningkatkan penjualannya.

Masalah lainnya berupa kesulitan para penjual/penentu keputusan untuk memilih paket produk yang akan dipasarkan sesuai dengan keragaman konsumen. Data penjualan yang berupa transaksi barang yang telah dibeli dan keinginan konsumen yang besar, harusnya

dapat diolah agar menghasilkan sebuah informasi yang berguna sebagai dasar untuk menentukan strategi apa yang cocok agar penjualan semakin meningkat. Menariknya, kehadiran teknologi informasi seakan menjadi alat bantu yang cocok dan tepat dalam menentukan keputusan bagi para pembuat keputusan ataupun penjual. Dengan menggunakan pendekatan/metode yang cocok diharapkan dapat menjadi jalan keluar bagi penjual dalam mengatasi permasalahan tersebut. Metode yang dapat diterapkan di dalam penelitian ini adalah *market basket analysis*. *Market basket analysis* merupakan salah satu metode di dalam *data mining* yang menentukan produk-produk yang dibeli konsumen secara bersamaan berdasarkan dari data transaksi konsumen tersebut [2].

Analisis keranjang pasar telah banyak digunakan di perusahaan sebagai alat untuk menemukan asosiasi produk. Produk-produk yang memiliki hubungan bisa disatukan dalam paket penjualan. Beberapa aspek yang dipelajari di *market basket analysis* antara lain profil minat pelanggan dengan minat produk untuk pemasaran[3], pola pembelian di lingkungan toko yang menjual berbagai macam *item* barang untuk meningkatkan penjualan[4]. Di dalam *market basket analysis*, poin penting yang menjadi acuan para pengambil keputusan adalah *association rules*. *Association rules* diturunkan dari *frequent itemset* menggunakan *support* dan *confidence* sebagai *threshold levels*. Aturan asosiasi yang dihasilkan sangat tergantung pada nilai *confidence* dari sebuah *itemset*.

Beberapa algoritma yang menggunakan *frequent itemset* antara lain Apriori[5], FP-Growth[6], Eclat[7] dan K-Apriori[8]. Khusus di dalam penelitian ini menggunakan



Gambar 1. Tahapan penelitian

algoritma FP-Growth. Pemilihan algoritma FP-Growth mengacu pada penelitian sebelumnya yang dilakukan oleh Widiastuti dan Sofi [9]. Mereka membandingkan dua buah algoritma yakni Apriori dan FP-Growth untuk mengukur waktu proses pada 2500 data transaksi. Hasilnya algoritma FP-Growth memerlukan waktu 1 jam 20 menit sedangkan Apriori memerlukan waktu 3 jam 12 menit dengan minimum *support* 40%. Beberapa penelitian serupa dengan penelitian ini seperti yang dilakukan oleh Nugroho, dkk [10] tentang rekomendasi pemilihan SMA menggunakan metode *naive bayes*. Hasil dari penelitian mereka adalah berupa daftar sekolah yang diurutkan berdasarkan probabilitas tertinggi dari hasil perhitungan dan prioritas favorit dari masing-masing sekolah. Penelitian yang dilakukan Virrayani dan Sutikno [11] tentang prediksi penjualan barang menggunakan metode *Adaptive Neuro-Fuzzy Inference System* (ANFIS). Hasil dari penelitian mereka yakni nilai MAPE (*Mean Absolute Percentage Error*) untuk pengujian seluruh produk di bawah batas toleransi *error* yakni 20%. Artinya metode ANFIS cukup baik digunakan untuk memprediksi penjualan barang.

Sementara itu, tujuan dari penelitian ini adalah memberikan rekomendasi bagi para penjual/*retailer* dalam memberikan paket penjualan barang bagi konsumennya, sedangkan manfaat dari penelitian ini adalah membantu para penjual untuk memilih strategi yang tepat dalam pemasaran produk sehingga berdampak pada peningkatan penjualan.

2. Metode Penelitian

Adapun tahapan atau proses dari penelitian ini dapat dilihat pada Gambar 1. Pada Gambar 1 dapat dikelompokkan dalam empat kategori besar, yakni:

1. Studi Awal

Pokok permasalahan di dalam penelitian ini adalah kesulitan penjual untuk menentukan paket produk yang akan ditawarkan kepada konsumen. Data yang digunakan adalah data transaksi penjualan di minimarket XYZ.

2. Proses

Sebelum dilakukan proses perhitungan dengan menggunakan metode FP-Growth, data transaksi penjualan di minimarket XYZ harus dilakukan pra-proses berupa mengisi *missing value*. Kemudian dilakukan transformasi data. Hasil dari transformasi data dapat dilihat pada Tabel 1.

Pada Tabel 1, terdapat 15 transaksi penjualan barang kebutuhan pokok di minimarket XYZ. Dari 15 transaksi tersebut, ada 3 transaksi dari konsumen yang hanya membeli 1 produk saja. Selanjutnya, data transaksi tersebut akan dilakukan *association rule mining*.

]

Tabel 1. Data transaksi minimarket XYZ

Kode Transaksi	Nama Barang
TRX001	Minyak Goreng, Gula, Kopi, Teh, Susu
TRX002	Air Mineral, Susu, <i>Snack</i>
TRX003	Buku, Pensil
TRX004	Permen, Gula
TRX005	Pensil
TRX006	<i>Snack</i> , Air Mineral
TRX007	Gula, Kopi
TRX008	Susu, Teh
TRX009	Gula, Buku, Pulpen
TRX010	Air Mineral, Minyak Goreng, Kopi
TRX011	<i>Snack</i> , Gula, Kopi, Buku, Pulpen
TRX012	Buku
TRX013	Permen, <i>Snack</i>
TRX014	Kopi, Teh, Susu
TRX015	Susu

a. Association Rule Mining

Analisis asosiasi atau *association rule mining* adalah teknik *data mining* untuk menemukan aturan asosiatif antara suatu kombinasi item. Analisis asosiasi dikenal juga sebagai salah satu teknik *data mining* yang menjadi dasar dari salah satu teknik *data mining* lainnya. Secara khusus, salah satu tahap analisis asosiasi yang menarik perhatian banyak peneliti untuk menghasilkan algoritma yang efisien, yaitu analisis pola frekuensi tinggi (*frequent pattern mining*) [12][7][6][13].

Analisis pola frekuensi tinggi. Tahap ini mencari kombinasi item yang memenuhi syarat minimum dari nilai *support* dalam *database*. Nilai *support* sebuah *item* diperoleh dengan persamaan (1). Sedangkan nilai *support* dari 2 *item* diperoleh dari persamaan (2).

$$\text{support (A)} = \frac{\text{jumlah transaksi mengandung A}}{\text{Total transaksi}} \quad (1)$$

$$\text{support (A} \cap \text{B)} = \frac{\text{jumlah transaksi mengandung A dan B}}{\text{total transaksi}} \quad (2)$$

Pembentukan aturan asosiatif. Setelah semua pola frekuensi tinggi ditemukan, kemudian dicari aturan asosiatif yang memenuhi syarat minimum untuk *confidence* dengan menghitung *confidence* aturan asosiatif $A \rightarrow B$. Nilai *confidence* dari aturan $A \rightarrow B$ diperoleh dari persamaan (3)

$$\text{confidence P(A|B)} = \frac{\text{jumlah transaksi mengandung A dan B}}{\text{jumlah transaksi mengandung A}} \quad (3)$$

Tabel 2. Frekuensi semua item barang untuk semua transaksi

Kode	Item Barang	Frekuensi
1	Gula	5
2	Kopi	5
3	Teh	3
4	Susu	5
5	Air Mineral	3
6	Snack	4
7	Buku	4
8	Minyak Goreng	2
9	Permen	1
10	Pulpen	2
11	Pensil	2

Tabel 3. Frekuensi item produk yang mengikuti minimum support count

Kode	Item Barang	Frekuensi
2	Gula	5
3	Kopi	5
4	Teh	3
5	Susu	5
6	Air Mineral	3
7	Snack	4
8	Buku	4

Tabel 4. Urutan item produk berdasarkan frekuensi terbesar

Kode	Item Barang	Frekuensi	Prioritas
2	Gula	5	1
3	Kopi	5	2
5	Susu	5	3
7	Snack	4	4
8	Buku	4	5
4	Teh	3	6
6	Air Mineral	3	7
Jumlah		29	

3. Hasil dan Pembahasan

Sebelum melihat hasil perhitungan dari sistem yang dibangun dengan menggunakan bahasa pemrograman Java, terlebih dahulu dilakukan perhitungan secara manual. Untuk itu ada beberapa langkah yang harus ditempuh yakni :

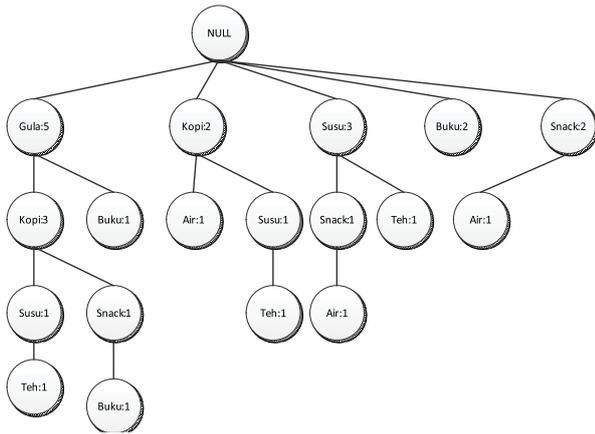
a. Pembuatan Frequent Pattern (FP) Tree

Untuk mencari *frequent itemset* dan *association rule* menggunakan algoritma FP-Growth ada beberapa langkah-langkah sebagai berikut:

- 1) Hitung terlebih dahulu *minimum support*
Asumsi *minimum support* adalah 20%, maka *minimum support count* adalah $(20\% * 15) = 3$, maka *minimum support count* adalah 3. Karena *minimum support count* adalah 3, maka ada item barang yang harus dibuang. Untuk lebih jelasnya dapat dilihat pada Tabel 2. Pada Tabel 2, di baris 8 hingga baris 11, frekuensi item barang dibawah *minimum support count* yang telah ditetapkan yakni 3. Itu berarti, *item* produk minyak goreng, permen, pulpen, dan pensil harus dihilangkan.
- 2) Pembuatan *Header Item*
Pembuatan *header* item mengikuti *minimum support count* yang telah ditentukan di awal. *Minimum support count* yang telah ditentukan adalah 3. Item barang yang memiliki frekuensi < 3, akan dibuang dari *header list*. Untuk lebih jelasnya ditunjukkan pada Tabel 3. Pada Tabel 3 terlihat ada 4 baris yang telah hilang. Yakni baris 1, 9, 10 dan 11. Produk tersebut tidak diikutkan di dalam proses berikutnya. Jadi produk yang ikut pada tahapan berikutnya adalah produk yang memiliki *minimum support count* ≥ 3 .
- 3) Prioritas item yang memiliki frekuensi terbesar
Setelah menghilangkan item produk yang di bawah *minimum support count*, langkah selanjutnya adalah memilih item prioritas yang memiliki frekuensi terbesar seperti yang ditunjukkan pada Tabel 4. Pada Tabel 4 terlihat produk gula, kopi, dan susu yang memiliki frekuensi paling banyak yaitu 5. Produk tersebut menempati urutan 1, 2, dan 3. Frekuensi paling sedikit ada pada produk air mineral yakni 3, sehingga mendapatkan prioritas terendah.
- 4) Susun *item* berdasarkan prioritas
Setelah frekuensi tertinggi dan terendah ditentukan, *item* tadi kembali diurukan berdasarkan prioritas. Di awal proses, data transaksi penjualan berjumlah 15. Hasil *order* (susun) ditunjukkan pada Tabel 5.

Tabel 5. Hasil order item produk berdasarkan prioritas

Kode Transaksi	Kode Barang	Hasil Order
TRX001	2,3,4,5	2,3,4,5
TRX002	6,5,7	5,7,6
TRX003	8	8
TRX004	2	2
TRX006	7,6	7,6
TRX007	2,3	2,3
TRX008	5,4	5,4
TRX009	3,8	3,8
TRX010	6,3	3,6
TRX011	7,2,3,8	2,3,7,8
TRX012	8	8
TRX013	7	7
TRX014	3,4,5	3,5,4
TRX015	5	5



Gambar 2. Pembuatan FP Tree

Tabel 6. Frequent itemset dengan 1 dan 2-itemsets

Pattern	Item	Min Support
Pattern 0	6	0.2
Pattern 1	4	0.2
Pattern 2	8	0.3
Pattern 3	7	0.3
Pattern 4	5	0.4
Pattern 5	3	0.4
Pattern 6	2	0.4
Pattern 7	4 5	0.2
Pattern 8	2 3	0.2

Tabel 7. Rule yang dihasilkan

Pattern	Min Support	Min Confidence
3 → 2	0.3	0.6
2 → 3	0.3	0.6
5 → 4	0.3	0.6
4 → 5	0.3	1

```
input_fp_growth.txt - Notepad
File Edit Format View Help
1 2 3 4 5
6 5 7
8 10
11 2
10
7 6
2 3
5 4
2 8 9
6 1 3
7 2 3 8 9
8
11 7
3 4 5
5
```

Gambar 3. Item barang diubah menjadi angka

```
public static void main(String [] arg) throws IOException{
    //String input = fileToPath("contextIGB.txt");
    String input = fileToPath("input_fp_growth.txt");
    String output = "../output.txt";

    // By changing the following lines to some other values
    // it is possible to restrict the number of items in the antecedent and
    // consequent of rules
    int maxConsequentLength = 40;
    int maxAntecedentLength = 40;
```

Gambar 4. Baris perintah untuk penyisipan inputan di FP-Growth

```
F HeaderList: [2, 3, 5, 7, 8, 4, 6]
-1 (count=1)
 2 (count=5)
  3 (count=3)
    5 (count=1)
      4 (count=1)
        7 (count=1)
          8 (count=1)
            8 (count=1)
              5 (count=3)
                7 (count=1)
                  6 (count=1)
                    4 (count=1)
                      8 (count=2)
                        7 (count=2)
                          6 (count=1)
                            3 (count=2)
                              6 (count=1)
                                5 (count=1)
                                  4 (count=1)
```

Gambar 5. Hasil pembentukan FP tree

```
----- FREQUENT ITEMSETS -----
L0
L1
pattern 0: 6 support : 3
pattern 1: 4 support : 3
pattern 2: 8 support : 4
pattern 3: 7 support : 4
pattern 4: 5 support : 5
pattern 5: 3 support : 5
pattern 6: 2 support : 5
L2
pattern 7: 4 5 support : 3
pattern 8: 2 3 support : 3
-----
```

Gambar 6. Hasil pembentukan frequent itemset

Pattern	#SUP:	#CONF:
3 ==> 2	3	0.6
2 ==> 3	3	0.6
5 ==> 4	3	0.6
4 ==> 5	3	1

Gambar 7. Hasil association rule

Pada Tabel 5, jika diperhatikan dengan saksama ada 1 transaksi yang hilang yaitu TRX005. Kode transaksi ini tidak dimasukkan dikarenakan *item* produk dari transaksi ini hanya pensil. Sedangkan produk pensil sendiri memiliki *minimum support count* di bawah 3.

- 5) Bentuk FP tree dari tiap transaksi. Tahap selanjutnya adalah membuat FP (*frequent pattern*) tree dari item transaksi. Hasil FP Tree seperti yang ditunjukkan pada Gambar 2. Pada Gambar 2 pohon *frequent pattern* dihasilkan dari semua *item* produk yang ada pada Tabel 5. *Item* produk seperti minyak goreng, pulpen, pensil, dan permen tidak ada di pohon tersebut. Gambar pohon juga tidak *absolute* atau wajib seperti gambar di atas. Yang terpenting adalah panah atau garis dan frekuensinya harus sama dengan dengan yang ada di Tabel 4.

6) Validasi

Setelah pohon FP (*frequent pattern*) terbentuk, kita terlebih dahulu harus melakukan validasi apakah pohon FP yang dibuat cocok dengan frekuensi yang ada pada tabel prioritas *item*. Tabel 4 memiliki jumlah frekuensi sebesar 29. Sedangkan pada Gambar 2, jumlah frekuensi yang ada di pohon FP berjumlah 29.

b. Penerapan Algoritma FP-Growth

Setelah *FP tree* terbentuk dan telah tervalidasi sesuai dengan jumlah frekuensi setiap *item*, maka langkah selanjutnya adalah penerapan algoritma FP-Growth.

- Langkah 1, menentukan *frequent itemset* menggunakan algoritma FP-Growth. Dari sekian banyak *frequent itemset* yang dihasilkan diambil item dari pola yang memiliki *minimum support (min support)* ≥ 0.3 atau 30%. *Frequent itemset* yang terbentuk ada 9 yang terbagi atas 1-*itemset* dan 2-*itemset* yang ditunjukkan pada Tabel 6.

$Pattern\ 0 = Support\ (Air\ Mineral) = 3 \div 14 = 0.21$
pembulatan ke atas (*ceiling*) menjadi 0.2.

$Pattern\ 1 = Support\ (Teh) = 3 \div 14 = 0.21$ pembulatan ke atas (*ceiling*) menjadi 0.2.

$Pattern\ 2 = Support\ (Buku) = 4 \div 14 = 0.28$
pembulatan ke atas (*ceiling*) menjadi 0.3.

$Pattern\ 3 = Support\ (Snack) = 4 \div 14 = 0.28$
pembulatan ke atas (*ceiling*) menjadi 0.3.

$Pattern\ 4 = Support\ (Susu) = 5 \div 14 = 0.35$
pembulatan ke atas (*ceiling*) menjadi 0.4.

$Pattern\ 5 = Support\ (Kopi) = 5 \div 14 = 0.35$
pembulatan ke atas (*ceiling*) menjadi 0.4.

$Pattern\ 6 = Support\ (Gula) = 5 \div 14 = 0.35$
pembulatan ke atas (*ceiling*) menjadi 0.4.

$Pattern\ 7 = Support\ (Teh)\ Support\ (Susu) = 3 \div 14 = 0.21$

$Pattern\ 8 = Support\ (Gula) \cap Support\ (Kopi) = 3 \div 14 = 0.21$.

Untuk lebih jelasnya dapat dilihat pada Tabel 6.

Pada Tabel 6 ada 9 buah *pattern* yang dihasilkan dari *frequent itemset*. Namun karena telah ditentukan *minimum support* ≥ 0.3 atau 30%. Maka *pattern* yang tersisa tinggal 9.

- Langkah 2, pembentukan *association rule* dari *frequent itemset* yang telah terbentuk. Dari 9 *frequent itemset*, *association rule* yang dihasilkan ada 4 seperti yang ditunjukkan pada Tabel 7.

Pada Tabel 7 ada 4 *association rule* (aturan) yang terbentuk yakni *item* kopi dan gula dapat menjadi satu paket dengan *minimum support* sebesar 0.3 dan *minimum confidence* sebesar 0.6 begitu juga produk gula dan kopi. Produk lainnya yakni teh dan susu dapat menjadi satu paket dengan *minimum support* sebesar 0.3 dan *minimum confidence* sebesar 0.6.

c. Perhitungan dengan Aplikasi

Aplikasi FP-Growth ini dibangun dengan menggunakan bahasa pemrograman Java. Bahasa ini dipilih karena merupakan bahasa pemrograman berbasis objek. Selain itu, Java juga dapat dijalankan di semua sistem operasi. Langkah-langkah menggunakan aplikasi FP-Growth adalah sebagai berikut:

- Ubah nama *item* barang menjadi deretan angka, misalnya gula diberi kode 2, dan seterusnya seperti

yang ditunjukkan pada Gambar 3.

Pada Gambar 3, aplikasi yang dibuat tidak dapat mengolah data berupa *string* atau kalimat. Jika setiap produk harus dikonversikan ke dalam nilai *integer* (bilangan bulat). Misalnya item gula diberi nilai setara dengan 2.

- Inputan item barang disimpan dalam ekstensi .txt. File berekstensi .txt kemudian disimpan di dalam *folder* yang digabung dalam satu *package* dengan aplikasi FP-Growth seperti yang ditunjukkan pada Gambar 4. Pada Gambar 4 ada sejumlah baris perintah di bahasa pemrograman Java untuk membuka file berekstensi .txt. Di baris berikutnya baris perintah untuk menampilkan *output* dari hasil perhitungan di algoritma FP-Growth.
- Hasil dari perhitungan algoritma FP-Growth akan menghasilkan pohon *frequent pattern*, *frequent itemset*, dan *association rule*.
- FP *tree* untuk dataset ini ditunjukkan pada Gambar 5. Pada Gambar 5 terlihat *header list* yang menampilkan *item* barang yang sudah mengikuti *minimum support count* $\geq 30\%$. Baris berikut terlihat ada indentasi atau baris menjorok ke dalam. Baris yang menjorok tersebut mewakili anak (*child*) dari *parent* di atasnya. Jika digambar akan persis sama seperti yang ditunjukkan pada Gambar 2.
- Frequent itemset* ditunjukkan pada Gambar 6. Pada Gambar 6 yang merupakan hasil *frequent itemset* yang dihasilkan dari aplikasi yang telah dibuat. *Frequent itemset* juga dibagi dalam 2 yakni *frequent itemset* 1 *item* dan 2 *item*. Jika melihat dari Tabel 6 memang sedikit ada selisih *minimum support*. Hal ini terjadi karena pembulatan antara perhitungan manual dan perhitungan dengan aplikasi berbeda.
- Hasil *association rule* untuk *dataset* ini ditunjukkan pada Gambar 7. Dari Gambar 7 dapat diinterpretasikan hasil *association rule* sebagai berikut.
 - Rule* 1, jika membeli kopi maka membeli gula dengan tingkat kepercayaan mencapai 60% dan didukung oleh 30% dari data keseluruhan.
 - Rule* 2, jika membeli gula maka membeli kopi dengan tingkat kepercayaan mencapai 60% dan didukung oleh 30% dari data keseluruhan.
 - Rule* 3, jika membeli susu maka membeli teh dengan tingkat kepercayaan mencapai 60% dan didukung oleh 30% dari data keseluruhan.
 - Rule* 4, jika membeli teh maka membeli susu dengan tingkat kepercayaan mencapai 100% dan didukung oleh 30% dari data.

Jika dipersempit, ditemukan dua pasangan *item* barang yakni kopi, gula dan teh, susu yang memiliki *support* sebesar 30% dan *confidence* sebesar 70%.

4. Diskusi

Pengembangan aplikasi berbasis *desktop* untuk memberikan rekomendasi yang tepat bagi para penjual atau para penentu keputusan. Penggunaan metode FP-Growth dalam menentukan pilihan pasangan *item* produk sangat tepat. Seperti pasangan *item* produk gula dan kopi yang memang saling membutuhkan. Jika ingin membuat

kopi, pastinya membuat gula untuk membuat rasa kopi menjadi manis. Didukung dengan nilai *minimum confidence* yang berada di level 60% seperti yang ditunjukkan pada Gambar 7 membuat pasangan dari *item* produk gula dan kopi bisa direkomendasikan untuk dijual secara bersama-sama. Pilihan produk lain seperti teh dan susu juga dapat dijual secara bersama-sama.

Fokus dari algoritma ini adalah memberikan pilihan pasangan *item* produk berdasarkan hasil transaksi penjualan. Para pengguna aplikasi ini tidak hanya digunakan oleh penjual, tetapi juga para penentu keputusan yang mempunyai kebijakan yang dapat diterapkan di tempat usahanya. Kekurangan dari penelitian ini adalah aplikasi masih menggunakan *desktop*, sehingga kesulitan untuk dapat diakses oleh banyak orang dari berbagai tempat. *Association rule* yang didapatkan masih belum dikonversikan ke dalam barang, sehingga para penjual/penentu keputusan harus dapat mengingat nilai bilangan bulat yang sesuai dengan nama barang.

5. Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah sistem yang dibangun telah dapat memberikan rekomendasi paket produk yang sesuai dengan keragaman keinginan konsumen. Penentuan rekomendasi paket produk dihitung berdasarkan tingkat frekuensi *item* yang dibeli oleh konsumen dengan memperhatikan *minimum support* yang telah ditetapkan. Aplikasi rekomendasi paket produk menggunakan metode FP-Growth dapat menjadi alternatif bagi para penentu keputusan/penjual untuk memilih produk yang bisa digabungkan di dalam satu paket yang diharapkan berdampak pada peningkatan penjualan.

6. Daftar Pustaka

- [1] E. Supriyani, "Pengaruh Biaya Pelaksanaan Promosi Melalui Pameran terhadap Tingkat Volume Penjualan pada PT Astra Internasional Tbk Isuzu Cabang Bogor," *J. Ilm. Ranggad.*, vol. 4, no. 1, pp. 69–74, 2004.
- [2] A. Muzakir and L. Adha, "Market Basket Analysis (MBA) Pada Situs Web E-Commerce Zakiyah Collection," *J. SIMETRIS*, vol. 7, no. 2, pp. 459–466, 2016.
- [3] S. . Weng and J. L. Liu, "Feature-based recommendations for one-to-one marketing," *Expert Syst. Appl.*, vol. 26, pp. 493–508, 2004.
- [4] Y.-L. Chen, K. Tang, R.-J. Shen, and Y.-H. Hu, "Market basket analysis in a multiple store environment," *Decis. Support Syst.*, vol. 40, no. 2, pp. 339–354, 2004.
- [5] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," *Proc. 20th VLDB Conf.*, pp. 487–499, 1994.
- [6] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," *Data Min. Knowl. Discov.*, vol. 8, no. 1, pp. 53–87, 2004.
- [7] H. Jiawei, K. Micheline, and P. Jian, *Data Mining Concepts and Techniques*, Third Edit. USA: Morgan Kaufmann Publishers, 2012.
- [8] L. C. Annie and A. Kumar, "Frequent Item set mining for Market Basket Data using K-Apriori algorithm," *Int. J. Comput. Intell. Informatics*, vol. 1, no. 1, pp. 14–18, 2011.
- [9] D. Widiastuti and N. Sofi, "Analisis Perbandingan Algoritma Apriori dan FP-Growth Pada Transaksi Koperasi," *UGJ*, vol. 8, no. 1, pp. 21–24, 2014.
- [10] Y. S. Nugroho, T. D. Salma, and S. Rokhanudin, "Implementasi Data Warehouse dan Data Mining Untuk Pengembangan Sistem Rekomendasi Pemilihan SMA," *Khazanah Informatika.*, vol. 2, no. 2, pp. 49–56, 2016.
- [11] A. Virrayyani and Sutikno, "Prediksi Penjualan Barang Menggunakan Metode Adaptive Neuro-Fuzzy Inference System (ANFIS)," *Khazanah Informatika.*, vol. 2, no. 2, pp. 57–63, 2016.
- [12] C. Zhang and S. Zhang, *Association Rule Mining: Models and Algorithms*. Heidelberg: Springer-Verlag Berlin, 2002.
- [13] Kusriani and E. T. Luthfi, *Algoritma Data Mining*. Yogyakarta: Andi Publisher, 2009.

Sistem Pendukung Keputusan dengan Metode AHP untuk Penilaian Kompetensi *Soft Skill* Karyawan

Rusydi Umar*, Abdul Fadlil, Yuminah

Program studi Magister Teknik Informatika

Universitas Ahmad Dahlan, Yogyakarta

Yogyakarta 55164

*rusydi_umar@rocketmail.com

Abstrak-Karyawan dalam perusahaan merupakan sumber daya utama yang dituntut untuk mampu memberikan pelayanan terbaik dan kinerja yang optimal. *Soft skill* karyawan adalah keterampilan individu karyawan yang dapat menunjang hubungan individu karyawan dengan karyawan lain, meningkatkan kinerja dan membuka prospek kariernya. Penilaian kinerja karyawan yang dilaksanakan oleh perusahaan umumnya hanya untuk penilaian prestasi kerja yakni bagaimana pekerjaan dapat dikerjakan dengan baik, mencapai target yang ditetapkan dan meraih tujuan akhir yang diinginkan (*hard skill*). Penilaian terkait kemampuan *soft skill* karyawan belum banyak dilakukan. Ada beberapa kriteria yang diterapkan beberapa perusahaan dalam melakukan penilaian kompetensi *soft skill*, tetapi kriterianya masih berbeda-beda. Penelitian ini membahas penilaian kompetensi *soft skill* karyawan dengan menerapkan empat kriteria. Keempat kriteria ini adalah kemampuan komunikasi, kemampuan bekerja sama, kejujuran, dan kemampuan interpersonal. Analisis data menerapkan metode *Analytical Hierarchical Process* (AHP), yang memungkinkan perhitungan matematis dengan berbagai kriteria. Hasil penelitian menunjukkan nilai rasio konsistensi 0.053 yang berarti kurang dari nilai rasio konsistensi yang digunakan dalam metode AHP yaitu 0.1, sehingga hasil perhitungan tersebut valid, dan dapat digunakan. Penelitian ini menghasilkan penilaian prioritas kompetensi *soft skill* yang dibutuhkan perusahaan sebagai berikut: Komunikasi 48%, Kerja sama 27%, Kejujuran 16 %, dan interpersonal 10%. Hasil penelitian ini membuktikan bahwa metode AHP dapat digunakan pada penilaian kompetensi *soft skill* karyawan.

Kata Kunci: *analytical hierarchical process*; sistem pendukung keputusan; penilaian kompetensi; *soft skill*

1. Pendahuluan

Mencapai pengembangan karier terbaik merupakan satu harapan besar bagi seseorang, karena akan memiliki implikasi luas pada kesejahteraan hidup. Kenyataan di lapangan menunjukkan bahwa *soft skill* memiliki peranan yang menentukan keberhasilan karyawan dalam pengembangan dirinya untuk mencapai kesuksesan. *Hard skill* memiliki peranan sebagai prasyarat minimal untuk karyawan dalam memasuki dunia kerja. Kompetensi *hard skill* adalah kompetensi yang berkaitan dengan kemampuan menyerap ilmu pengetahuan atau keahlian dan merupakan kemampuan untuk melakukan tugas atau kegiatan tertentu. *Soft skill* berkaitan dengan kemampuan karyawan dalam berinteraksi dengan sesama karyawan baik dalam pekerjaan maupun di luar pekerjaan

Soft skill dalam dunia kerja dapat menjadi penunjang kompetensi teknis atau *hard skill*. Karyawan yang mempunyai kompetensi teknis dan keahlian profesi yang baik tetapi tidak mempunyai kompetensi *soft skill* yang baik maka dapat menghambat kinerja karyawan tersebut. Kompetensi-kompetensi *soft skill* yang dibutuhkan di dunia kerja sudah teridentifikasi dengan jelas. Rilman dkk. [1] menyebutkan bahwa *soft skill* yang seharusnya dimiliki karyawan adalah kemampuan komunikasi, kemampuan menyelesaikan masalah, kemampuan dalam tim, kemampuan belajar sepanjang hayat, kemampuan mengelola informasi, kemampuan wirasaha, kemampuan beretika, moral dan professional, dan kemampuan kepemimpinan.

No.	Kemampuan/keterampilan	Skor Kepentingan
1.	Kemampuan Komunikasi	4,69
2.	Kejujuran/Integritas	4,59
3.	Kemampuan Bekerja Sama	4,54
4.	Kemampuan Interpersonal	4,50
5.	Beretika	4,46
6.	Motivasi/Inisiatif	4,42
7.	Kemampuan Beradaptasi	4,41
8.	Daya Analitik	4,36
9.	Kemampuan Komputer	4,21
10.	Kemampuan Berorganisasi	4,05
11.	Berorientasi pada Detail	4,00
12.	Kepemimpinan	3,97
13.	Kepercayaan Diri	3,95
14.	Ramah	3,85

Sumber: Hasil Survei NACE USA (2002).

Gambar 1. Kompetensi *Soft Skill* yang diperlukan dunia kerja. Sumber Hasil Survei NACE USA [1]

Kemampuan-kemampuan tersebut, untuk lebih jelas dapat dilihat dalam gambar 1.

Perusahaan dalam kurun waktu tertentu selalu melakukan penilaian kinerja karyawannya. Penilaian yang dilakukan setiap perusahaan mengikuti kebijakan yang berbeda-beda. Penilaian kinerja karyawan tersebut sebagai bentuk penilaian kerja nyata atas standar kualitas maupun kuantitas yang dihasilkan oleh karyawan. Penilaian prestasi kerja penting bagi setiap karyawan maupun perusahaan untuk mengambil keputusan dan menentukan kebijaksanaan selanjutnya. Kegagalan perusahaan dalam

menilai kinerja karyawan dapat mengakibatkan kegagalan perusahaan dalam mencapai target umum perusahaan dan dapat menurunkan motivasi karyawan dalam berprestasi.

Sistem Pendukung Keputusan (SPK) merupakan sistem yang dapat memberikan pemecahan masalah, melakukan komunikasi untuk pemecahan masalah tertentu dengan terstruktur maupun tidak terstruktur. SPK didesain untuk dapat digunakan dan dioperasikan dengan mudah oleh orang yang hanya memiliki kemampuan dasar pengoperasian komputer. SPK dibuat dengan menerapkan adaptasi kompetensi yang tinggi sehingga dapat dijadikan sebagai alternatif dalam pengambilan sebuah keputusan [2]. SPK terdiri atas dua kata kunci, yaitu sistem informasi dan keputusan. Sistem informasi merupakan serangkaian prosedur formal dengan tahapan di mana data dikelompokkan, diproses sehingga menghasilkan informasi yang selanjutnya diberikan kepada pengguna. Keputusan adalah serangkaian kegiatan untuk memilih suatu tindakan dalam memecahkan masalah. Tindakan memilih dari alternatif yang dihadapi yang didasarkan pada fakta dan dilakukan melalui pendekatan sistematis yang dapat memberikan solusi terbaik yang dilakukan oleh manajer disebut pengambilan keputusan [3].

Penelitian terkait SPK telah banyak dilakukan. Palasara meneliti pengambilan keputusan untuk menilai karyawan terbaik dengan kriteria antara lain kuantitas kerja, kualitas kerja, kejujuran, tanggung jawab, motivasi diri, dedikasi, penampilan, keterampilan kerja, inisiatif, pemanfaatan waktu, kerja sama, komunikasi, dan disiplin [4]. Penelitian ini menghasilkan nilai indeks konsistensi sebesar 0.01 artinya nilai kesalahan di bawah 10%, dan membuktikan bahwa metode AHP cocok digunakan untuk pemilihan karyawan terbaik.

Purnomo, dkk. membandingkan metode AHP, TOPSIS, dan AHP-TOPSIS dalam memutuskan penerima beasiswa [5]. Mereka menguji 4 kriteria yang meliputi nilai UASBN, nilai TPA, nilai IQ, dan nilai piagam. Penelitian tersebut menghasilkan bahwa metode TOPSIS menjadi metode terbaik.

Permana menerapkan SPK untuk memilih sekolah menengah kejuruan favorit dengan menggunakan kriteria fasilitas, biaya, kualitas, dan SDM Siswa [6]. Mereka menggunakan metode *multi criteria decision* dan menghasilkan bahwa teknik komputer dan jaringan adalah jurusan yang terfavorit di SMK.

Penelitian lain terkait sistem pendukung keputusan adalah [7] yang mencermati sebuah rancangan analisis sistem untuk penilaian *soft skill* siswa SMK sebagai calon tenaga kerja. Penelitian [8] menerapkan sistem pendukung keputusan untuk memprediksi stok barang. Sementara itu penelitian [9] menggunakan SPK untuk pemilihan rumah tinggal dan penelitian [10] menerapkan sistem pakar untuk melakukan diagnosis gangguan layanan telekomunikasi pada sebuah operator telekomunikasi Indonesia. Sedangkan penelitian [11] menghasilkan sistem untuk menangani tambahan belajar mengajar peminatan sesuai dengan kegiatan pembelajaran yang dibutuhkan.

Penelitian ini mengkaji tentang penilaian kompetensi *soft skill* karyawan dengan metode AHP, sedangkan pada penelitian yang dilaporkan di atas banyak mengkaji penilaian karyawan terbaik pada kompetensi *hard skill* saja. Berdasarkan 14 kebutuhan kompetensi *soft skill* yang banyak dibutuhkan oleh dunia kerja, maka penelitian ini

mengambil 4 kriteria teratas yaitu: kemampuan komunikasi, kemampuan bekerja sama, kejujuran, dan interpersonal. Kenyataan di lapangan bahwa 4 kompetensi tersebut memang sangat menentukan kesuksesan karyawan dalam bekerja. Empat kompetensi tersebut saling terkait antara satu dengan yang lain, misalnya jika seorang karyawan mempunyai kemampuan komunikasi yang baik, maka karyawan tersebut mempunyai tingkat kejujuran yang baik, dari kejujuran yang baik maka akan dapat bekerja sama dalam tim dengan baik, dengan demikian karyawan tersebut mempunyai kemampuan interpersonal yang baik [12]. Kompetensi *soft skill* mempunyai peran yang amat sangat penting, karena menyangkut kemampuan dasar (*basic skill*) seseorang dalam menyelesaikan suatu pekerjaan [13].

Sebagai bentuk usaha untuk mendorong kinerja perusahaan, semakin banyak perusahaan yang melakukan penilaian kinerja karyawan menitikberatkan pada kemampuan *soft skill*, maka akan memberikan kontribusi nyata terhadap pembentukan karakter kehidupan bangsa. Dengan demikian kemampuan *soft skill* sangat menentukan kemajuan diri dan dapat terwujud dengan melakukan pembiasaan di kehidupan sehari-hari. Sekarang bekerja juga harus memperhatikan pengembangan *soft skill* (interaksi sosial) sebab hal ini sangat penting dalam pembentukan karakter karyawan sehingga mampu bersaing, beretika, bermoral, sopan santun, dan berinteraksi dengan masyarakat. Sehingga terbentuklah karakter yang baik. Karakter yang baik akan berimbas pada kinerja yang baik dan terbentuklah karyawan yang baik. Sudah saatnyalah perusahaan-perusahaan melakukan penilaian kinerja karyawan dari segi kemampuan *soft skill* yang dimiliki.

2. Dasar Teori

Analytical Hierarchical Process (AHP) merupakan hierarki dengan *input* atau masukan utama berupa pandangan manusia. Dikembangkan oleh Prof. Thomas Lorie Saaty dari Wharton Business School awal tahun 1970. Metode ini digunakan untuk mencari urutan atau ranking prioritas dari berbagai alternatif dalam pemecahan masalah [14]. AHP banyak digunakan untuk mengekspresikan pengambilan suatu keputusan yang sangat efektif dari suatu permasalahan yang kompleks.

Penentuan prioritas dengan metode AHP dilakukan melalui beberapa tahapan yaitu:

- a. Menyusun hierarki.
- b. Menilai kriteria dan alternatif.
- c. Memilih prioritas.
- d. Menentukan nilai konsistensi logis.

Di dalam mengambil keputusan, penting untuk diketahui baik tidaknya nilai konsistensi yang digunakan. Adapun langkah-langkah yang harus dilakukan adalah:

- 1) Menghitung nilai perbandingan yang sudah ditetapkan dengan membagi nilai skala yang ada di setiap sel dibagi dengan nilai sel prioritas.
- 2) Hasil perhitungan dari langkah no. 1 di masing-masing sel dibagi dengan jumlah total di masing-masing kolom.
- 3) Mencari Eigen dengan cara menghitung rata-rata per baris.

- 4) Mencari Lamda (λ) dengan cara mengalikan masing-masing nilai eigen per baris dengan jumlah total per kolom.
- 5) Mencari lamda (λ maks) dengan cara menjumlahkan hasil lamda
- e. Menentukan nilai indeks konsistensi (CI)
Rumus yang digunakan sebagai berikut:

$$CI = \frac{\lambda_{max} - n}{n - 1}$$

Keterangan:

n = banyaknya kreteria.

- f. Menentukan rasio konsistensi (CR)
Rumus yang digunakan sebagai berikut:

$$CR = \frac{CI}{RI}$$

Keterangan:

CR = Rasio Konsistensi

CI = Indeks Konsistensi

RI = Indeks Random Konsistensi

- g. Memeriksa konsistensi hierarki.
Jika hasil perhitungan nilai rasio konsistensi lebih dari 10%, maka harus diperbaiki atau dihitung ulang. Tapi jika rasio konsistensi kurang atau sama dengan 0,1 maka dapat dinyatakan benar nilai perhitungannya. Nilai RI atau *index random* seperti terlihat pada tabel 1.

3. Metode

a. Pengumpulan Data

Metode deskriptif analitik yang akan digunakan dalam penelitian ini dengan menyajikan hasil survey yang berupa kuesioner. Data sekunder diambil melalui berbagai media, seperti: internet, buku literatur dan jurnal serta artikel-artikel sehingga didapatkan informasi yang akurat mengenai penilaian kompetensi *soft skill*. Selanjutnya data diolah dengan menggunakan pendekatan proses hierarki analitis (AHP) untuk menentukan kriteria kompetensi *soft skill* yang akan digunakan.

b. Tahap Penelitian

Tahapan penelitian dalam penelitian ini adalah dimulai dengan menentukan masalah, pemilihan kriteria, komputasi AHP, perhitungan analisis data, kesimpulan (lihat gambar 2).

c. Menentukan Masalah dan Pemilihan Kriteria

Sasaran yang ingin dicapai dalam penelitian ini adalah mendapatkan cara penilaian kompetensi *soft skill*. Kriteria penilaian dipilih dan diuji menggunakan metode AHP sesuai dengan kebutuhan dunia kerja seperti terlihat pada Tabel 2.

Kriteria-kriteria pada Tabel 2 dimasukkan ke dalam tabel perbandingan dua kriteria berpasangan seperti terlihat pada tabel 3.



Gambar 2. Tahap-tahap penelitian

Tabel 1. Nilai Indeks Random

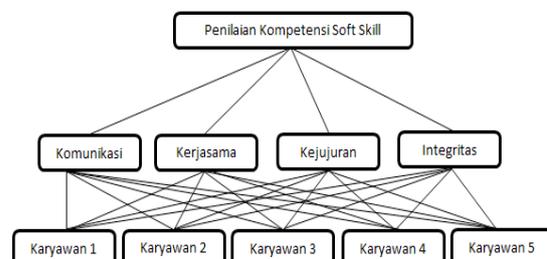
N	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49

Tabel 2. Sasaran dan Kriteria

Sasaran	Kriteria
Penilaian Kompetensi <i>soft skill</i> Karyawan	Kemampuan Komunikasi
	Kejujuran
	Kemampuan Bekerjasama
	Kemampuan Interpersonal

Tabel 3. Perbandingan Antar 2 Kriteria

Kriteria	Perbandingan	Kriteria
Komunikasi	2	Interpersonal
Kerja sama	3	Kejujuran
Kejujuran	4	Kerjasama
Interpersonal	5	Komunikasi



Gambar 3. Hierarki metode AHP penilaian *soft skill*

Tabel 4. Kriteria dalam Skala Penilaian *Soft Skill*

Kriteria	Komunikasi	Kerjasama	Kejujuran	Interpersonal
Komunikasi	1	2	3	5
Kerjasama	0.50	1	1.50	4
Kejujuran	0.33	0.67	1	1.67
Interpersonal	0.2	0.4	0.6	1
Jumlah	2.03	4.07	6.1	11.67

Tabel 5. Hasil perhitungan Lamda maks Penilaian *Soft Skill*

Kriteria	Komunikasi	Kerjasama	Kejujuran	Interpersonal	EIGIN	λ (lamda)	λ maks
komunikasi	0.491803279	0.49180328	0.49	0.428571429	0.475995316	0.9678571	4.144941452
kerjasama	0.245901639	0.24590164	0.25	0.342857143	0.270140515	1.0985714	
kejujuran	0.163934426	0.16393443	0.16	0.142857143	0.158665105	0.9678571	
interpersonal	0.098360656	0.09836066	0.10	0.085714286	0.095199063	1.1106557	

Tabel 3 menunjukkan bahwa kemampuan komunikasi dan kemampuan interpersonal memiliki perbandingan dengan skala 2, yang berarti nilai keduanya sama atau setara. Kemampuan kerja sama dengan kejujuran berada pada skala 3 yang berarti kemampuan kerja sama sedikit lebih penting daripada kejujuran. Perbandingan antara kejujuran dengan kemampuan kerja sama berada pada skala 4 berarti nilai di antara keduanya sama. Kemampuan interpersonal dengan komunikasi berada pada skala 5 yang berarti kemampuan interpersonal lebih penting daripada komunikasi.

d. Komputasi AHP

Komputasi AHP dilakukan melalui tahapan-tahapan sebagai berikut.

a. Menyusun hierarki penilaian *soft skill*

Gambar 3 memperlihatkan hierarki penilaian *soft skill* yang menunjukkan bahwa posisi tingkat 1 adalah sasaran. Sasaran yang dimaksud adalah Penilaian *soft skill* karyawan. Posisi tingkat 2 adalah kriteria yang meliputi komunikasi, kerja sama, kejujuran, dan interpersonal. Sedangkan posisi tingkat 3 adalah sebagai alternatif yaitu karyawan yang dinilai.

b. Penilaian kriteria dan menentukan prioritas

Kriteria dan skala penilaian yang digunakan seperti terlihat pada tabel 4.

Tabel 5 merupakan tabel perbandingan dengan masing-masing kriteria berdasarkan skala yang sudah ditetapkan pada tabel 4, sebagai contoh komunikasi dibandingkan dengan komunikasi akan menghasilkan nilai 0 maksudnya $1/1=0$, komunikasi dibandingkan dengan kerjasama akan menghasilkan 0.50 maksudnya $1/2=0.50$, dilanjutkan perbandingan hingga semua kriteria.

c. Konsistensi Logis

Dalam menentukan nilai konsistensi dilakukan dengan langkah-langkah sesuai pada metode AHP, dan diperoleh hasil seperti terlihat pada tabel 5

Tabel 6 memperlihatkan hasil perhitungan:

$$\text{Eigen} = 0.491+0.491+0.49+0.428 = 0.475$$

$$\lambda (\text{lamda}) = 0.475 \times 2.03 = 0.967$$

λ maks = $0.967+1.098+0.967+1.110= 4.144$
sehingga penghitungan *Consistency Index* (CI) sebagai berikut:

$$\begin{aligned} CI &= \frac{\lambda_{\max} - n}{n - 1} \\ &= \frac{4.144941452 - 4}{4 - 1} \\ &= 0.048313817 \end{aligned}$$

Dan penghitungan Rasio Konsistensi/*Consistency Ratio* (CR) sebagai berikut:

$$\begin{aligned} CR &= \frac{CI}{RI} \\ &= \frac{0.048313817}{0.9} \\ &= 0.053682019 \end{aligned}$$

d. Memeriksa konsistensi hierarki

Jika nilainya Rasio Konsistensi lebih dari 10%, maka penilaian *data judgment* harus diperbaiki. Namun jika Rasio Konsistensi (CI/CR) kurang atau sama dengan 0,1 maka benar. Dari hasil perhitungan diperoleh nilai Rasio Konsistensi 0.053682019 adalah kurang dari atau sama dengan 0.1 maka hasil perhitungan bisa dinyatakan benar. Sehingga diperoleh prioritas kompetensi *soft skill* yang dibutuhkan perusahaan sebagai berikut; komunikasi 48%, kerja sama 27%, kejujuran 16 %, dan interpersonal 10%.

4. Hasil

Perhitungan dilakukan pada program aplikasi pengolah angka Microsoft Excel 2010, sampai menghasilkan prioritas penilaian kompetensi *soft skill* karyawan. Hasil yang diperoleh dari penelitian yang dilakukan dapat disajikan sebagai berikut:

Tabel 6. Hasil Uji Validitas Kompetensi Komunikasi

NAMA	KRITERIA KOMUNIKASI					
	K1	K2	K3	K4	K5	K6
rx _y	0.252	0.857	0.662	0.85	0.894	0.837
r-hitung	0.712	2.424	1.873	2.405	2.528	2.368
r-tabel	0.632	0.632	0.632	0.632	0.632	0.632
signifikansi validitas	valid	valid	valid	valid	valid	valid

Tabel 7. Hasil Uji Validitas Kompetensi Kerja Sama

Nama	Kriteria Kerjasama						
	B1	B2	B3	B4	B5	B6	B7
rx _y	0.537	0.879	0.836	0.792	0.885	0.508	0.883
r-hitung	1.52	2.485	2.363	2.241	2.504	1.437	2.496
r-tabel	0.632	0.632	0.632	0.632	0.632	0.632	0.632
signifikansi validitas	valid	valid	valid	valid	valid	valid	valid

Tabel 8. Hasil Uji Validitas Kompetensi Kerja Sama

Nama	Kriteria Kejujuran					
	J1	J2	J3	J4	J5	J6
rx _y	0.553	0.616	0.853	0.828	0.853	0.754
r-hitung	1.565	1.743	2.412	2.341	2.412	2.134
r-tabel	0.632	0.632	0.632	0.632	0.632	0.632
signifikansi validitas	valid	valid	valid	valid	valid	valid

Tabel 9. Hasil Uji Validitas Kompetensi Interpersonal

Nama	Kriteria Interpersonal							
	I1	I2	I3	I4	I5	I6	I7	I8
rx _y	0.539	0.563	0.613	0.757	0.83	0.623	0.363	0.53
r-hitung	1.524	1.594	1.734	2.142	2.346	1.763	1.026	1.499
r-tabel	0.632	0.632	0.632	0.632	0.632	0.632	0.632	0.632
signifikansi validitas	valid	valid	valid	valid	valid	valid	valid	valid

Tabel 10. Nilai Reliabilitas Alpha Cronbach's

Kriteria	Alpha Cronbach's	keterangan
Komunikasi	0.798	Reliabel
Kerja sama	0.987	Reliabel
Kejujuran	0.836	Reliabel
interpersonal	0.604	Reliabel

Tabel 11. Hasil Rekap Kuisisioner Karyawan

Karyawan	komunikasi	kerjasama	kejujuran	interpersonal
K1	18	23	21	26
K2	24	20	30	36
K3	21	25	24	26
K4	18	24	22	28
K5	22	26	30	33
K6	27	30	24	33
K7	20	22	24	28
K8	27	32	30	35
K9	22	26	27	29
K10	26	32	25	32

Tabel 12. Contoh Perhitungan Matriks Perbandingan pada Kriteria Kamunikasi

Karyawan	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10
K1	1	0.75	0.8571429	1	0.818181818	0.6666667	0.9	0.66666667	0.818182	0.692307692
K2	1.333333333	1	1.1428571	1.333333333	1.090909091	0.8888889	1.2	0.88888889	1.090909	0.923076923
K3	1.166666667	0.875	1	1.166666667	0.954545455	0.7777778	1.05	0.77777778	0.954545	0.807692308
K4	1	0.75	0.8571429	1	0.818181818	0.6666667	0.9	0.66666667	0.818182	0.692307692
K5	1.222222222	0.916666667	1.047619	1.222222222	1	0.8148148	1.1	0.814814815	1	0.846153846
K6	1.5	1.125	1.2857143	1.5	1.227272727	1	1.35	1	1.227273	1.038461538
K7	1.111111111	0.833333333	0.952381	1.111111111	0.909090909	0.7407407	1	0.740740741	0.909091	0.769230769
K8	1.5	1.125	1.2857143	1.5	1.227272727	1	1.35	1	1.227273	1.038461538
K9	1.222222222	0.916666667	1.047619	1.222222222	1	0.8148148	1.1	0.814814815	1	0.846153846
K10	1.444444444	1.083333333	1.2380952	1.444444444	1.181818182	0.962963	1.3	0.962962963	1.181818	1
Jumlah	12.50	9.38	10.71	12.50	10.23	8.33	11.25	8.33	10.23	8.65

Karyawan	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	EIGEN
K1	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.080	0.080	0.08000
K2	0.106666667	0.11	0.1066667	0.106666667	0.106666667	0.10666667	0.11	0.11	0.107	0.107	0.10667
K3	0.093333333	0.09	0.0933333	0.093333333	0.093333333	0.09333333	0.09	0.09	0.093	0.093	0.09333
K4	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.080	0.080	0.08000
K5	0.097777778	0.10	0.0977778	0.097777778	0.097777778	0.09777778	0.10	0.10	0.098	0.098	0.09778
K6	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.120	0.120	0.12000
K7	0.088888889	0.09	0.0888889	0.088888889	0.088888889	0.0888889	0.09	0.09	0.089	0.089	0.08889
K8	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.120	0.120	0.12000
K9	0.097777778	0.10	0.0977778	0.097777778	0.097777778	0.09777778	0.10	0.10	0.098	0.098	0.09778
K10	0.115555556	0.12	0.1155556	0.115555556	0.115555556	0.1155556	0.12	0.12	0.116	0.116	0.11556

Tabel 13. Nilai Matriks Perbandingan untuk semua kriteria dan eigen prioritas perusahaan.

Karyawan	Komunikasi	Kerjasama	Kejujuran	Interpersonal	EIGEN PRIORITAS
K1	0.08	0.08846154	0.0817121	0.08496732	0.475995316
K2	0.106666667	0.07692308	0.1167315	0.117647059	0.270140515
K3	0.093333333	0.09615385	0.0933852	0.08496732	0.158665105
K4	0.08	0.09230769	0.0856031	0.091503268	0.095199063
K5	0.097777778	0.1	0.1167315	0.107843137	
K6	0.12	0.11538462	0.0933852	0.107843137	
K7	0.088888889	0.08461538	0.0933852	0.091503268	
K8	0.12	0.12307692	0.1167315	0.114379085	
K9	0.097777778	0.1	0.1050584	0.094771242	
K10	0.115555556	0.12307692	0.0972763	0.104575163	

Tabel 14. Pemeringkatan Penilaian *soft skill* Karyawan

KARYAWAN	NILAI	URUTAN
K8	0.11954099	1
K6	0.11317512	2
K10	0.11311835	3
K5	0.10255155	4
K2	0.10245623	5
K9	0.09911522	6
K3	0.09301464	7
K7	0.08889639	8
K4	0.08523074	9
K1	0.08290078	10

a. Hasil Uji Validitas dan Reliabilitas

1) Uji Validitas

a) Kriteria Kompetensi Komunikasi

Validitas mempunyai arti sejauh mana ketepatan dan kecermatan alat ukur dalam melakukan fungsi alat ukurnya. Suatu kriteria dikatakan valid apabila kriteria tersebut mampu mencapai tujuan

pengukuran dari kontrak amatan dengan tepat [15]. Suatu kriteria atau pernyataan dikatakan valid apabila p value (signifansi hasil analisis) \leq taraf signifikansi (α) 0,05. Hasil uji validitas kriteria kemampuan komunikasi seperti terlihat pada tabel 6. Dari perhitungan yang tersaji di Tabel 6, diperoleh hasil r -hitung= 0,712. Selanjutnya

nilai tersebut dikonsultasikan dengan r -tabel *product moment* pada taraf signifikansi dengan $df = 8$, taraf signifikansi $5\% = 0,632$, dan taraf signifikansi $1\% = 0,765$. Jika r hitung $> r$ tabel taraf sig $1\% > r$ tabel taraf sig 5% maka dapat dinyatakan valid dan sebaliknya, perhitungan yang sama untuk semua *item* pertanyaan.

- b) Kriteria Kompetensi Kerjasama
Nilai korelasi hasil uji validitas untuk kriteria kompetensi kerja sama seperti terlihat pada Tabel 7.

Dari perhitungan Tabel 7 diperoleh hasil r -hitung = 1.52. Jika r hitung $> r$ tabel taraf sig $1\% > r$ tabel taraf sig 5% maka dapat dinyatakan valid dan sebaliknya, perhitungan yang sama untuk semua item pertanyaan.

- c) Kriteria Kompetensi Kejujuran
Nilai korelasi hasil uji validitas untuk kriteria kompetensi kejujuran seperti terlihat pada tabel 8.

Dari perhitungan Tabel 8 diperoleh hasil r -hitung = 1.565. Jika r hitung $> r$ tabel taraf sig $1\% > r$ tabel taraf sig 5% maka dapat dinyatakan valid dan sebaliknya, perhitungan yang sama untuk semua item pertanyaan.

- c) Kriteria Kompetensi Interpersonal
Nilai korelasi hasil uji validitas untuk kriteria kompetensi interpersonal seperti terlihat pada tabel 9.

Dari perhitungan Tabel 9, diperoleh hasil r -hitung = 1.524. Jika r hitung $> r$ tabel taraf sig $1\% > r$ tabel taraf sig 5% maka dapat dinyatakan valid dan sebaliknya, perhitungan yang sama untuk semua item pertanyaan.

- 2) Uji Reliabilitas

Hasil uji reliabilitas dapat diketahui dari nilai alpha cronbach's selengkapnya disajikan pada tabel 10.

Hasil uji reliabilitas menunjukkan bahwa nilai alpha cronbach's komunikasi 0.798, kerja sama 0.987, kejujuran 0.836, dan interpersonal 0.604. Karena nilai alpha cronbach's yang didapat masing-masing > 0.6 . Hal ini membuktikan tiap variabel memiliki reliabilitas yang tinggi atau memiliki konsistensi yang baik sebagai alat ukur.

b. Hasil Rekap Kuisisioner

Dari *form* kuisisioner yang dibagikan ke karyawan, dapat dihitung skor untuk masing-masing kriteria untuk masing-masing karyawan. Hasilnya seperti terlihat pada tabel 11.

Tabel 11 menunjukkan hasil rekap nilai dari masing-masing kriteria dengan skor yang dipakai, sebagai berikut:

- 1= Sangat Tidak Mampu
2= Tidak Mampu
3= Cukup Mampu
4= Mampu
5= Sangat Mampu

c. Matriks Perbandingan untuk Setiap Kriteria

Hasil rekap dilakukan perhitungan matriks perbandingan setiap karyawan untuk masing-masing kriteria, di bawah ini sebagai contoh hasil perhitungan matriks perbandingan pada kriteria komunikasi seperti terlihat pada tabel 12.

Contoh hasil perhitungan matriks perbandingan untuk kriteria komunikasi, dilakukan dengan cara membandingkan nilai masing-masing karyawan misalnya K1 nilainya 1 maka 1/1, K1/K2 maka 1/1.33, K1/K3 maka 1/1.16 semua kriteria dari semua karyawan, selanjutnya akan menghasilkan eigen masing-masing kriteria.

d. Menentukan Prioritas/Pemeringkatan

Untuk melakukan perhitungan prioritas atau pemeringkatan dari masing eigen setiap kriteria yang sudah didapat, seperti terlihat pada tabel 13.

Tabel 13 menunjukkan bahwa nilai prioritas untuk K1 diperoleh dari $(0.08 \times 0.475) + (0.08 \times 0.270) + (0.08 \times 0.158) + (0.08 \times 0.095) = 0.082$ dan seterusnya sampai dengan K10

Perhitungan matriks perbandingan semua kriteria dan nilai prioritas yang digunakan pada perusahaan menghasilkan pemeringkatan penilaian kompetensi *soft skill* karyawan, seperti terlihat pada tabel 14.

Tabel 14 menunjukkan bahwa karyawan yang memenuhi kriteria sesuai kebutuhan perusahaan dalam penilaian kompetensi *soft skill* dengan 3 prioritas tertinggi adalah K8 dengan nilai 0.11954, K6 dengan nilai 0.11317 dan K10 dengan nilai 0.11311. Terdapat angka yang tidak sama pada angka ke 5 di belakang koma. Yaitu pada data K10 dan K5. Pada penelitian sebelumnya yang berjudul sistem pengambilan keputusan pemilihan karyawan terbaik menggunakan metode *Analytic Hierarchy Process* [4] belum menunjukkan perbedaan yang jelas dalam penentuan prioritas karena masih terdapat data yang sama sampai digit ke 4 di belakang koma, yaitu pada peringkat ke 2 dan 3 dengan nilai sama-sama 0.1040 dan pada peringkat 6 dan 7 dengan nilai sama-sama 0.0980.

5. Kesimpulan

Hasil penelitian yang dilakukan dapat diambil kesimpulan bahwa uji reliabilitas menunjukkan bahwa nilai alpha cronbach's komunikasi 0.798, kerja sama 0.987, kejujuran 0.836, dan interpersonal 0.604. Nilai alpha Cronbach's yang didapat masing-masing > 0.6 . Hal ini membuktikan tiap variabel memiliki reliabilitas yang tinggi atau memiliki konsistensi yang baik sebagai alat ukur. Pengambilan keputusan penilaian kompetensi *soft skill* karyawan dengan metode AHP menghasilkan nilai *index consistency* sebesar 0.05 artinya nilai kesalahan di bawah 5 %. Sehingga nilai *index* konsistensinya benar dan dapat digunakan. Penelitian ini membuktikan bahwa metode AHP dapat digunakan dalam penilaian kompetensi *soft skill* karyawan sampai menentukan nilai prioritas karyawan tertinggi. Peneliti menyadari adanya ketidaksempurnaan dan saran yang dapat peneliti berikan untuk meningkatkan penelitian yang sama, yaitu dalam penentuan kriteria lebih banyak lagi dan juga menentukan subkriterianya, sehingga hasilnya akan lebih maksimal. Penelitian selanjutnya diharapkan melakukan pengujian dengan metode lain yang sejenis.

6. Daftar Pustaka

- [1] A. Rilman and A. Djamaris, "Analisis Faktor Kompetensi *Soft Skills* Mahasiswa yang Dibutuhkan Dunia Kerja Berdasarkan Persepsi Manajer dan HRD Perusahaan," *J. Manaj.*, vol. 17, no. 2, pp. 160–174, 2013.
- [2] A. Abdul Chamid and B. Surarso, "Implementasi Metode AHP dan Promethee untuk Pemilihan Supplier," *J. Sist. Inf. Bisnis Mei*, vol. 2, pp. 13–2015, 2015.
- [3] N. Komalasari, "Sistem Pendukung Keputusan Kelaikan Terbang (SPK2T)," *J. Ind. Elektro dan Penerbangan*, vol. 4, no. 1, 2018.
- [4] N. Palasara, "Sistem Pengambilan Keputusan Pemilihan Karyawan Terbaik Menggunakan Metode *Analytic Hierarchy Process*," vol. 4, no. 1, pp. 31–46, 2017.
- [5] E. N. Sejati Purnomo, S. Widya Sihwi, and R. Anggrainingsih, "Analisis Perbandingan Menggunakan Metode AHP, TOPSIS, dan AHP-TOPSIS dalam Studi Kasus Sistem Pendukung Keputusan Penerimaan Siswa Program Akselerasi," *J. Teknol. Inf. ITSmart*, vol. 2, no. 1, p. 16, 2016.
- [6] S. D. H. Permana, "Sistem Penunjang Keputusan Pemilihan Sekolah Menengah Kejuruan Teknik Komputer dan Jaringan yang Terfavorit dengan Menggunakan *Multi-Criteria Decision Making*," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 1, pp. 11–19, 2015.
- [7] S. Utaminingsih, "Pengembangan *Soft Skill* Berbasis Karier pada SMK di Kota Semarang," *Din. Pendidik.*, vol. 6, no. 2, pp. 119–133, 2016.
- [8] R. Menggunakan, M. Confussion, S. Prodi, T. Elektro, F. Teknik, and U. T. Madura, "Simposium Nasional Teknologi Terapan (SNTT) 5 2017 ISSN: 2339-028X," pp. 267–271, 2017.
- [9] H. Supriyono, "Pemilihan Rumah Tinggal Menggunakan Metode *Weighted Product*," *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 1, no. 1, pp. 23–28, 2015.
- [10] E. Lestari and E. U. Artha, "Sistem Pakar dengan Metode Dempster Shafer untuk Diagnosis Gangguan Layanan INDIHOME di PT TELKOM Magelang," *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 3, no. 1, pp. 16–24, 2017.
- [11] O. J. F. Wassalam, R. Umar, and A. Yudhana, "Implementasi dan Pengembangan Sistem *E-Learning* Berbasis Web pada STIMIK Muhammadiyah Paguyangan," in *Proceeding SENDI*, 2017, pp. 104–107.
- [12] M. U. Manara, "*Hard Skills* dan *Soft Skills* pada Bagian Sumber Daya Manusia di Organisasi Industri," *J. Psikol. Tabularasa*, vol. 9, no. 1, pp. 37–47, 2014.
- [13] A. Rokhayati, R. Kambara, and M. Ibrahim, "Pengaruh Soft Skill dan Perencanaan Karier terhadap Kinerja Karyawan dengan Kualitas Pelatihan Sebagai Variabel Modertor (Studi empiris pada PT. Krakatau Tirta Industri Cilegon)," *J. Ris. Bisnis dan Manaj. Tirtayasa*, vol. 1, no. 2, 2017.
- [14] A. Nugroho and S. Hartati, "Sistem Pendukung Keputusan Berbasis AHP (*Analytical Hierarchy Process*) untuk Penentuan Kesesuaian Penggunaan Lahan (Studi Kasus: Kabupaten Semarang)," *Sist. Pendukung Keputusan Berbas. Ahp (Analytical Hierarchy Process. Untuk Penentuan Kesesuaian Pengguna. Lahan (Studi*, vol. 9, no. 2, pp. 1–14, 2012.
- [15] Sugiyono, *Metode Penelitian*. Bandung: CV Alfabeta, 2000.

Peningkatan Kinerja Jaringan Komputer dengan *Border Gateway Protocol (BGP)* dan *Dynamic Routing* (Studi Kasus PT Estiko Ramanda)

Tati Ernawati*, Jemi Endrawan

Program Studi Teknik Informatika

Politeknik TEDC Bandung

Cimahi

*tatiernawati@yahoo.com

Abstrak- *Routing* dalam jaringan menjadi elemen utama karena merupakan proses pemilihan salah satu dari beberapa kemungkinan jalur pengiriman paket data dari alamat pengirim ke alamat tujuan. Protokol *routing* diperlukan untuk membentuk tabel *routing* berdasarkan informasi *routing* yang dipertukarkan setiap selang waktu tertentu sehingga pengalamatan pada paket data yang akan dikirim menjadi lebih jelas. Penelitian ini membangun dan mengkaji kinerja jaringan komputer (*computer network performance*) pada studi kasus di perusahaan yang bergerak di bidang jasa layanan internet, peningkatan kebutuhan jasa layanan internet menuntut adanya pengelolaan jaringan internal di perusahaan dalam upaya peningkatan kinerja jaringan komputer. Metode *comparison testing* digunakan dalam pengujian, parameter yang diuji adalah *latency* dan *traceroute* menggunakan *dynamic routing Border Gateway Protocol (BGP)*. Hasil uji menunjukkan sistem jaringan komputer dengan BGP memiliki efisiensi jaringan yang tinggi dengan rata-rata *latency* 0 ms (82 ms tanpa BGP) dan *traceroute* (konten lokal) 4 hop (8 hop tanpa BGP), namun *traceroute* (konten non lokal) memiliki nilai persentase yang sama dikarenakan seluruh *prefix* non lokal harus diperoleh dari *port backbone* (port lama tanpa BGP).

Kata Kunci: kinerja jaringan komputer, *dynamic routing*, *Border Gateway Protocol*, *latency*, *traceroute*

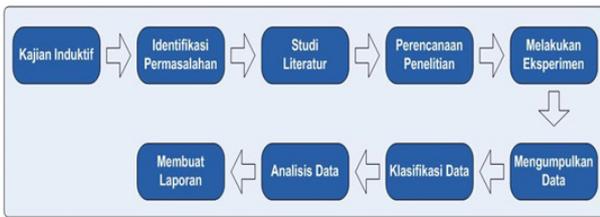
1. Pendahuluan

Koneksi antar jaringan (internet) terbentuk dari jaringan yang berbeda-beda. Supaya antar jaringan tersebut dapat saling berkomunikasi maka diperlukan teknik tertentu. Jaringan luas (*Wide Area Network = WAN*) menggunakan teknologi *Transmission Control Protocol/Internet Protocol (TCP/IP)* menggunakan nomor sebagai kode lokasi sehingga pengiriman paket data dapat sampai ke alamat yang dituju. TCP/IP membagi tugas masing-masing perangkat transmisi yaitu penerimaan paket data dan pengiriman paket data dalam jaringan sehingga jika terjadi permasalahan dalam pengiriman paket data dapat dipecahkan dengan baik [1].

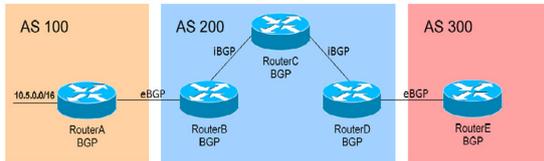
Routing pada jaringan menjadi teknik penting dalam transmisi data karena *routing* merupakan proses pemilihan salah satu dari beberapa kemungkinan jalur pengiriman paket data dari alamat pengirim ke alamat penerima/tujuan. Peta atau tabel *routing* digunakan *router* untuk mengetahui bagaimana meneruskan paket-paket ke alamat tujuan dengan menggunakan jalur terbaik [2]. Protokol *routing* diperlukan untuk membentuk tabel *routing* berdasarkan informasi *routing* yang dipertukarkan setiap

selang waktu tertentu sehingga pengalamatan pada paket data yang akan dikirim menjadi lebih jelas. Penggunaan teknik *routing* pada jaringan berskala besar seperti *dynamic routing* mengindikasikan bahwa terdapat dinamika dalam suatu jaringan. Jaringan komputer dapat mengalami gangguan pada *router* seperti terputusnya koneksi yang berakibat *router* lain harus melakukan *update* pada tabel *routing*.

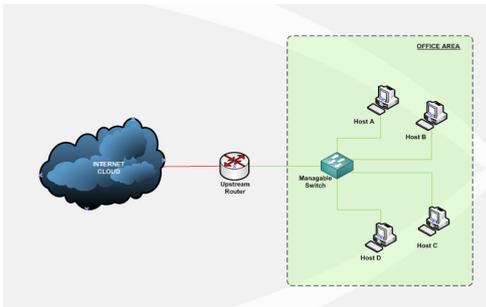
Terdapat beberapa penelitian yang telah dilakukan dengan topik yang sama terkait *routing* protokol BGP. Nabella dalam [3] mengkaji penerapan BGP dalam jaringan baik pada sistem otonom (*Autonomous System = AS*) yang sama maupun berbeda dengan atribut dalam pemilihan jalur pada jaringan yaitu *weight* dan *local preference*. I Gede Putra, dkk dalam [4] mendesain jaringan dengan tools *Graphical Network Simulator* menggunakan protokol *Internal Border Gateway Protocol (iBGP)* yang difokuskan pada analisis kebijakan dalam memanipulasi informasi *routing* mencakup *multipath*, *confederation*, *router reflector*, *next hop* dan *route map*. Caesar, dkk dalam [5] memaparkan tentang *policy routing* protokol BGP pada jaringan ISP, sementara itu Huston [6] menganalisis sistem tabel *routing* BGP pada internet.



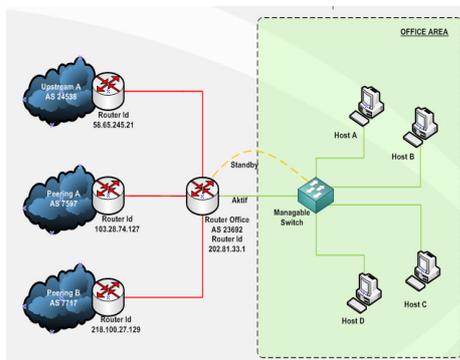
Gambar 1. Metode eksperimen [7]



Gambar 2. eBGP dan iBGP [9]



Gambar 3. Topologi Jaringan PT Estiko Ramanda yang berjalan



Gambar 4. Rancangan topologi Jaringan PT Estiko Ramanda dengan dynamic routing BGP

Namun, penulis belum menemukan penelitian yang mengambil parameter *latency* dan *traceroute* untuk mengamati kinerja jaringan komputer yang dibangun. Penulis mengambil studi kasus pada perusahaan yang bergerak di bidang jaringan internet (ISP) yaitu PT Estiko Ramanda berlokasi di Jakarta. Permasalahan yang dihadapi perusahaan yaitu kebutuhan konsumen akan jasa layanan internet semakin meningkat. Peningkatan kebutuhan tersebut menuntut adanya pengelolaan jaringan internal di perusahaan, dalam upaya peningkatan kinerja jaringan komputer (*computer network performance*). Berdasarkan permasalahan tersebut penulis melakukan kajian menggunakan metode *dynamic routing* BGP yang difokuskan kepada peningkatan kinerja jaringan yaitu *latency* dan *traceroute* yang dibandingkan dengan sistem yang sedang berjalan. BGP digunakan karena menjadi sebuah protokol *routing* yang dipandang mampu menjamin

performance routing antar jaringan yang berbeda AS, memiliki sifat *scalability* dan konvergensi yang mampu bekerja dengan baik dalam sebuah AS maupun antar AS.

Paper ini selanjutnya ditulis sebagai berikut. Bagian 2 menjelaskan metode dan berisi uraian tentang langkah-langkah dalam penelitian yang dilakukan. Bagian 3 menyajikan hasil penelitian berupa proses rancang bangun jaringan komputer menggunakan *dynamic routing* BGP. Bagian 4 memaparkan analisis dan diskusi berupa hasil pengujian terhadap kinerja jaringan yang difokuskan pada parameter *latency* dan *traceroute* serta perbandingan hasil penelitian penulis dengan hasil penelitian sebelumnya. Bagian 5 memberikan kesimpulan berdasarkan hasil kajian dan hasil uji.

2. Metode

Metodologi penelitian yang digunakan adalah metode eksperimen [7] yang terdiri dari 9 (sembilan) tahapan seperti gambar 1.

a. Kajian Induktif

Kajian induktif bertolak dari kaidah (hal-hal atau peristiwa) khusus untuk menemukan kaidah yang umum, berfokus kepada sebagian kecil dari fenomena yang relevan dengan permasalahan yang diamati. Kajian induktif dilakukan dengan mengkaji tentang BGP secara khusus yang merupakan bagian dari berbagai jenis *dynamic routing* secara umum.

b. Studi Literatur

Pada bagian ini akan dipaparkan beberapa konsep dasar terkait penelitian yang dikaji.

1) *Broadway Gateway Protocol* (BGP)

BGP adalah protokol *routing* standar yang bertujuan untuk memilih jalur-jalur *interdomain*. Fungsi utama dari BGP untuk mempertukarkan *network reachability information* antar suatu BGP *router* dengan BGP *router* yang lain. BGP tidak dibangun untuk rute dalam satu AS, tetapi dibangun untuk rute antar AS [8]. Informasi tersebut mencakup informasi jumlah AS yang berada dalam jalur penyampaian informasi. Terdapatnya informasi ini, dapat dibentuk grafik dari AS *path* yang saling terkoneksi sehingga dapat menghindari terjadinya *routing loop*. Selain fungsi di atas, BGP juga digunakan untuk menerapkan *policy routing* di tingkat *interdomain*.

2) Operasi BGP

Informasi *routing* dipertukarkan dengan membangun sebuah sesi yang berlandaskan pada koneksi TCP antar satu BGP *router* dengan BGP *router* yang lain [9]. Terdapat dua jenis hubungan BGP yaitu iBGP (*internal BGP*) yang berfungsi agar *router-router* internal mengetahui rute untuk mencapai suatu tujuan dan eBGP (*external BGP*) yang berfungsi untuk mengumumkan *reachable prefixes* ke BGP *router* tetangga [10]. Perbedaan iBGP dengan eBGP adalah bahwa iBGP tidak melakukan perubahan atribut AS *path*. Untuk menghindari *routing loops*, dalam satu AS koneksi antar BGP *router* dengan iBGP diterapkan topologi *full mesh*. Gambar 2 menunjukkan

contoh sesi iBGP dan eBGP.

Setelah semua *route* yang terbaik diumumkan ke BGP *router* tetangga, BGP *router* kemudian menangani kestabilan tabel *routing* yang dimilikinya. Apabila ada perubahan tabel *routing*, hanya informasi *update* yang diumumkan ke BGP *peer*-nya. BGP tidak mensyaratkan *refresh* tabel *routing* secara periodik oleh karena itu agar perubahan *policy* lokal dapat langsung diterapkan dengan benar tanpa perlu mereset sesi BGP, diperlukan kemampuan *route refresh* dari *router* BGP tersebut [10].

3) Pemilihan Rute dalam BGP

Terdapat 4 (empat) komponen utama dalam memahami proses pemilihan rute dalam BGP. Komponen pertama adalah *filter input* dan *output* yang dapat dikonfigurasi untuk setiap sesi BGP. Fungsi dari *filter* rute *router* atau dapat juga digunakan untuk memanipulasi atribut rute adalah untuk menolak rute yang diterima atau dikirim oleh tersebut. Proses *filter* ini dilakukan oleh operator jaringan. Komponen kedua adalah tabel *routing* BGP, tabel *routing* ini berisi semua rute yang diterima oleh *router* dan lolos dari proses *filter input*. Atribut dari *route* disimpan dalam tabel *routing* dan dilakukan proses *update* oleh *filter input*. Komponen ketiga adalah *decision process*, terjadi pemilihan rute terbaik dari rute-rute yang disimpan dalam tabel *routing* untuk setiap *prefix* tujuan. Apabila sebuah rute dipilih sebagai rute terbaik untuk suatu *prefix* tujuan, maka rute tersebut diinstal di tabel *forwarding* dan diumumkan ke *router* tetangga. Tabel *forwarding* merupakan komponen keempat dari *router*. Setiap paket yang diterima, tabel *forwarding* akan dilihat kemudian ditentukan *outgoing interface* yang digunakan untuk meneruskan paket ke tujuan.

4) Latency dan Traceroute

Latency dalam jaringan komputer didefinisikan sebagai banyaknya waktu yang dibutuhkan satu paket data untuk kembali ke pengirimnya dari satu titik ke titik yang lain. *Latency* bergantung pada kecepatan media transmisi dan penundaan transmisi oleh perangkat (misal *router* dan modem). *Latency* rendah mengindikasikan efisiensi jaringan yang tinggi [11].

Traceroute (*Tracer*) adalah perintah untuk menunjukkan rute yang dilewati paket untuk mencapai tujuan. Dilakukan dengan mengirim pesan *Internet Control Message Protocol* (ICMP) *Echo Request* ke tujuan dengan nilai *Time to Live* yang semakin meningkat. Rute yang ditampilkan adalah daftar *interface router* (yang paling dekat dengan *host*) yang terdapat pada jalur antara *host* dan tujuan [12].

c. Perencanaan Penelitian

Tahap ini ditentukan objek penelitian beserta pembuatan instrumen penelitian yang diperlukan. Objek penelitian ini mengambil studi kasus di PT Estiko Ramand, perusahaan yang bergerak di bidang penyediaan

jasa internet (*provider*) yang beralokasi di Jakarta. Instrumen penelitian dibuat untuk memperoleh data di lapangan yang dipergunakan untuk membangun dan menguji kinerja jaringan.

d. Eksperimen

Pelaksanaan eksperimen dilakukan untuk membangun sistem, menguji kebenaran cara kerja dan mengukur kinerja sistem jaringan yang difokuskan pada parameter *latency* dan *traceroute*.

e. Pengumpulan Data, Klasifikasi Data dan Analisis Data

Data diperoleh dari hasil eksperimen berupa data hasil uji kinerja jaringan. Berdasarkan hasil eksperimen data diklasifikasikan menjadi data kinerja jaringan yang sedang berjalan dan data kinerja jaringan yang dibangun (menggunakan protokol BGP) kemudian dilakukan analisis data. Analisis data dilakukan dengan komparasi kinerja jaringan hasil eksperimen sebelum dan sesudah implementasi BGP berdasarkan hasil uji, adapun data yang dianalisis mencakup parameter *latency* dan *traceroute*.

f. Identifikasi Permasalahan

Berdasarkan kajian induktif diidentifikasi permasalahan yang dikaji yaitu bagaimana membangun jaringan komputer dengan menggunakan protokol BGP dan bagaimana hasil uji jika dibandingkan dengan sistem jaringan yang sedang berjalan (*eksisting*).

3. HASIL

Hasil yang dicapai oleh peneliti adalah sistem jaringan komputer menggunakan *dynamic routing* BGP.

a. Tahap Perancangan Topologi Jaringan

Topologi jaringan ini berupa gambar *router-router* beserta konektivitas layer 3 antar *router-router* tersebut. Topologi yang digunakan berdasarkan pada topologi jaringan *ISP* yang dilengkapi *AS Number*. Topologi yang merepresentasikan kondisi jaringan PT Estiko Ramanda, pemodelan topologi hanya dilakukan pada beberapa *router* yang menjadi *ASBR* (*Autonomous System Border Router*) dan *router* yang mengendalikan pemilihan rute. Hasil dari pemetaan topologi jaringan PT Estiko Ramanda dapat dilihat pada gambar 3.

Pada gambar 4 topologi yang sekarang berjalan akan ditambahkan *router* yang berfungsi sebagai *router border* sekaligus *router* distribusi yang terkoneksi langsung dengan *managable switch*.

b. Tahap Identifikasi Kebutuhan Perangkat Jaringan

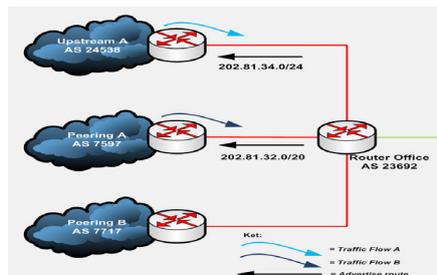
Perangkat jaringan menggunakan 1 unit *Router*, 1 unit *managable switch* dengan IP Public dan *Autonomous System Number* (*AS Number*). *Router* digunakan sebagai *router border* atau perangkat yang berhubungan langsung dengan *router upstream* atau *router* lawan sedangkan fungsi *managable switch* adalah sebagai perpanjangan tangan dari fungsi *router* sebagai perangkat distribusi ke arah *end user*. *Managable switch* digunakan karena perangkat tersebut dapat memisahkan *link* di layer 2 dengan penggunaan *Virtual LAN* (VLAN). Spesifikasi *router* seperti pada tabel 1.

Tabel 1. Spesifikasi Perangkat Jaringan

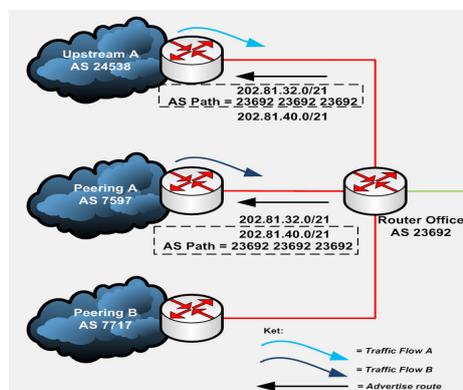
No	Perangkat Keras		
	Jenis	Type	Spesifikasi
1	Mikrotik Routerboard	CCR1016-12G	- Tileria Tile-Gx16 1.2GHz 16 Cores - RAM 2GB - 12 LAN Gigabit <i>port</i>
2	Cisco Catalyst	WS-C3750G-24TS-S1U	- 24 Ethernet 10/ 100/1000 <i>ports</i> - 4 SFP-based Gigabit Ethernet <i>ports</i>

Tabel 2. IP Address

No IP	IP Address		
	Subnet	Jumlah Host	Status
202.81.32.0	/24	254	Tidak di Advertise
202.81.33.0	/24	254	Tidak di Advertise
202.81.34.0	/24	254	Advertise ke Upstream 1
202.81.35.0	/24	254	Advertise ke Upstream 1
202.81.36.0	/24	254	Advertise ke Upstream 1
202.81.37.0	/24	254	Advertise ke Upstream 1
202.81.38.0	/24	254	Advertise ke Upstream 1
202.81.39.0	/24	254	Advertise ke Upstream 1
202.81.40.0	/24	254	Advertise ke Upstream 1
202.81.41.0	/24	254	Advertise ke Upstream 1
202.81.42.0	/24	254	Advertise ke Upstream 1
202.81.43.0	/24	254	Advertise ke Upstream 1
202.81.44.0	/24	254	Advertise ke Upstream 1
202.81.45.0	/24	254	Advertise ke Upstream 2
202.81.46.0	/24	254	Advertise ke Upstream 2
202.81.47.0	/24	254	Advertise ke Upstream 2



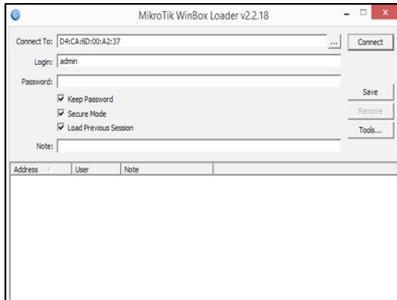
Gambar 5. Load balancing dengan selective advertisement



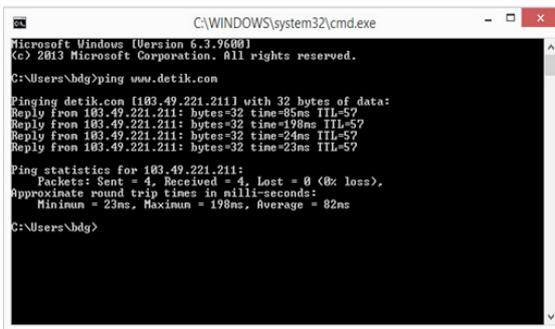
Gambar 6. Load Balancing Incoming Traffic dengan AS Path Prepending

Tabel 3. Spesifikasi Koneksi Jaringan

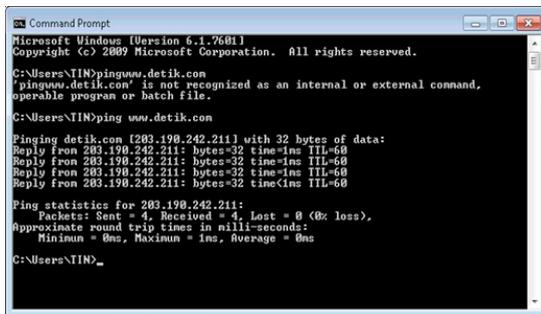
Konektivitas				
No	Jenis	Kecepatan	Status	Keterangan
1	Upstream	140 Mbps	Aktif	Non lokal
2	IIX	1 Gbps	Belum Aktif	Lokal



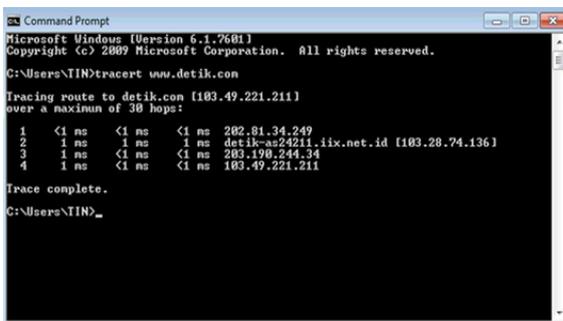
Gambar 7. Winbox Interface



Gambar 8. Latency konten lokal sebelum BGP



Gambar 9. Latency konten lokal sesudah BGP



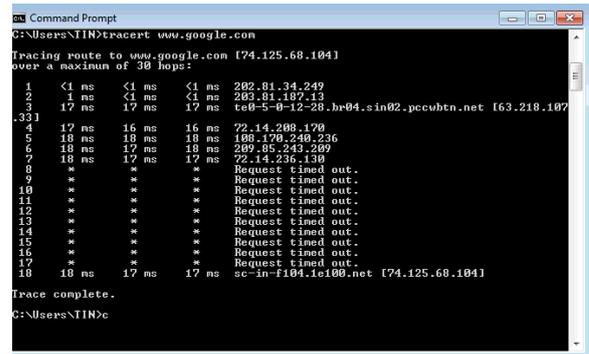
Gambar 10. Traceroute konten lokal sesudah BGP

c. Tahap Konfigurasi dan Implementasi Sistem Jaringan

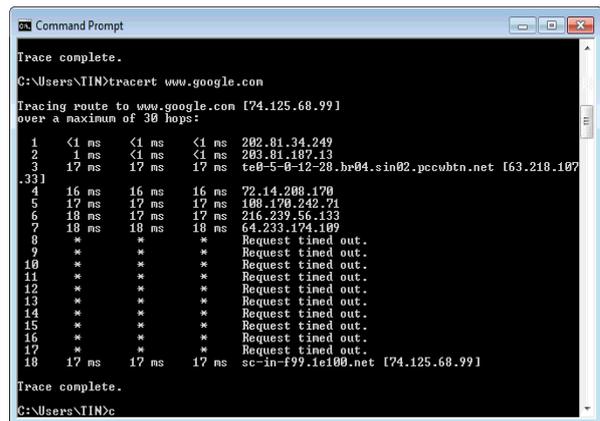
IP Public dan AS Number diperoleh dari Asosiasi Penyelenggara Jasa Internet Indonesia (APJII). IP Address dan spesifikasi koneksi jaringan pada Tabel 2 dan Tabel 3.



Gambar 11. Traceroute konten lokal sebelum BGP



Gambar 12 Traceroute Content Non Lokal Sebelum BGP



Gambar 13. Traceroute Content Non Lokal Sesudah BGP

Berdasarkan topologi jaringan yang dibuat dilakukan konfigurasi sebagai berikut:

- 1) Interkoneksi fisik antar perangkat.
Pemasangan interkoneksi dilakukan dengan menghubungkan *interface* fisik dari semua perangkat sesuai dengan gambar 3. Kabel *Unshielded Twisted-Pair* (UTP) digunakan dalam membangun jaringan keseluruhan interkoneksi
- 2) Interkoneksi *virtual*.
Pengalokasian IP *Address Intranet* dilakukan oleh administrator IP sedangkan untuk kebutuhan *interdomain* IP *Address* ditentukan dari persetujuan pihak *upstream provider*.
- 3) Konfigurasi *intranet*.
Konfigurasi ini dilakukan pada *managable* switch dengan menggunakan *Virtual Local Area Network* (VLAN).
- 4) Sesi-sesi BGP.
Pada tahapan ini akan dibangun sesi-sesi BGP antar *router*, pada jaringan yang akan dibuat ini

akan dibangun 3 sesi BGP yaitu dengan pihak *upstream provider*, *peering A* dan *peering B*.

5) BGP *policy*.

Policy diterapkan pada setiap sesi BGP antar *router*. Konfigurasi diperlukan dalam *traffic engineering*, karena pada konfigurasi ini dilakukan rekayasa *routing* maupun *bandwidth*. BGP *Policy* mengendalikan *outgoing traffic* maupun *incoming traffic*. Atribut *Local_Pref* digunakan pada pengendalian *outgoing traffic* yaitu dengan menentukan nilai dari *Local_Pref* di setiap sesi BGP. Sesi BGP dengan nilai *Local_Pref* yang lebih besar akan lebih dipilih dibandingkan dengan *Local_Pref* yang lebih kecil. Nilai *default Local_Pref* adalah 100.

Router Office melakukan *advertise* prefix 202.81.32.0/20 ke arah *Peering A* dan *advertise* prefix 202.81.34.0/24 ke arah *upstream A*. Tujuannya *incoming traffic* menuju ke prefix 202.81.34.0/24 dapat melalui jalur *traffic flow A*, sedangkan menuju *host* tujuan yang masih dalam prefix 202.81.32.0/20 dapat melalui *traffic flow B* sehingga dicapai kondisi *loadbalance incoming traffic*. Tuning BGP dapat menyebabkan kegagalan sistem apabila terjadi masalah pada jalur antara *router office* dengan *upstream A*, akibatnya seluruh *host* selain dalam prefix 202.81.34.0/24 tidak akan dikenali oleh internet, untuk menghindarinya prefix 202.81.32.0/20 harus dilakukan *advertise* juga ke arah *router peering A*.

Teknik lain untuk mengendalikan *incoming traffic* dengan *As_Path prepending*. Gambar 6, pembagian prefix 202.81.32.0/20 menjadi dua bagian yaitu prefix 202.81.32.0/21 dan 202.81.40.0/21. Prefix 202.81.32.0/20 dilakukan *advertise* ke arah *router peering A* tanpa *As_Path prepending* dan prefix 202.81.40.0/21 dengan *As_Path prepending* sebanyak 3(tiga) kali. Sebaliknya, prefix 202.81.32.0/21 dilakukan *advertise* ke arah *router upstream A* dengan *As_Path prepending* 3 kali sedang prefix 202.81.40.0/21 tanpa *As_Path prepending*. Hal ini dilakukan untuk *loadbalance* trafik yang masuk ke AS 23692. Jalur *traffic flow A* akan menjadi jalur utama untuk trafik menuju prefix 202.81.40.0/21 dan jalur *backup* untuk prefix 202.81.32.0/21; berlaku sebaliknya untuk jalur *traffic flow B*. Interkoneksi menggunakan IIX yang merupakan produk APJII dimana semua anggota dapat melakukan *share* dan akses *content* dengan anggota lainnya sehingga mempersingkat rute yang dibutuhkan ke arah *content* tujuannya. Pada tahapan ini akan diaktifkan koneksi *peering* dengan IIX sehingga PT Estiko Ramanda mempunyai 2 kaki sebagai jalur *backbone*.

Konfigurasi BGP jaringan menggunakan mikrotik dilakukan dengan menggunakan *command line interface* atau winbox. Tampilan awal winbox dapat dilihat pada gambar 7.

4. Diskusi

a. Pengujian

Pengujian dilakukan dengan metode *comparison testing* yaitu dengan mengambil data yang masuk dan keluar pada *port trunk* yang mengarah ke jaringan internal. Data yang masuk dan keluar dibandingkan antara data sebelum BGP diimplementasikan dan setelah BGP diimplementasikan. Parameter yang diuji mencakup *latency* dan *traceroute*.

b. Hasil Pengujian

Pengujian dengan *ping test* dilakukan untuk uji *latency* ke arah IP tujuan dan *traceroute* untuk memperlihatkan rute yang dilalui ke arah IP tujuan. Situasi saat BGP belum diimplementasikan pada jaringan diperlihatkan di gambar 8, yang menunjukkan *latency ping* ke arah konten lokal dengan nilai rata-rata sebesar 82 ms. Ketika BGP telah diimplementasikan, *latency* turun ke angka rata-rata sebesar 0 ms (lihat gambar 9). Sebagai konsekuensi penurunan *latency*, kecepatan akses ke arah konten lokal lebih cepat setelah BGP diimplementasikan pada jaringan.

Gambar 10 memperlihatkan hasil *traceroute* ke arah konten lokal ketika BGP sudah diimplementasikan pada jaringan. Gambar menunjukkan rute ke arah konten tersebut terdapat 4 hop atau 4 router yang dilalui untuk mencapai IP tujuan.

Gambar 11 memperlihatkan hasil *traceroute* ketika BGP belum diimplementasikan pada jaringan. Gambar memperlihatkan bahwa untuk mencapai IP tujuan, paket data harus melewati rute yang terdiri dari 8 hop atau 8 router.

Gambar *capture* yang telah disajikan menunjukkan bahwa setelah implementasi BGP trafik ke arah konten lokal akan berpindah melalui rute yang lebih pendek atau sedikit. Untuk akses ke konten non lokal trafik masih melewati *port backbone* lama karena seluruh prefiks non lokal hanya didapatkan dari *port backbone* lama. Gambar 12 dan gambar 13 memperlihatkan rute yang dilalui paket data ketika dilakukan akses terhadap konten non lokal. Rute yang sama ditempuh meskipun BGP sudah diimplementasikan sebab *port backbone* 2 atau *port* baru tidak mendapatkan prefiks ke arah non lokal.

c. Diskusi

Hasil pengujian yang telah disajikan menunjukkan bahwa *routing* BGP menghasilkan kinerja (*performance*) jaringan komputer lebih baik. Hasil senada diperoleh oleh beberapa peneliti lain yang mengamati kinerja *routing* protokol BGP.

Hasil penelitian [3] menyimpulkan bahwa BGP menjamin jaringan bebas *loop*. Analisis terhadap atribut *weight* menunjukkan terjadinya *advertise* ke *router* apabila ada perangkat yang memiliki nomor AS sama. Pada jaringan eksternal, bobot tertinggi dipilih sebagai jalur terbaik untuk mencapai tujuan, dan apabila mengalami gangguan koneksi maka jalur lain dijadikan jalur *back-up*. *Local-preference* digunakan pada *router* internal dengan nomor AS yang sama, bobot *local-preference* tertinggi menjadi jalur terbaik yang dipilih untuk mencapai alamat tujuan.

Hasil penelitian [4] membuktikan bahwa kinerja *router* BGP stabil. *Routing* mampu mencegah *failure link* secara lebih baik, dan terdapat skalabilitas pada pembagian beban. Sementara hasil penelitian [5] menggarisbawahi pentingnya mekanisme kebijakan *routing* untuk mengatasi permasalahan implementasi BGP bagi ISP. Selanjutnya penelitian [13] menunjukkan bahwa BGP bersifat *scalable*, artinya implementasi dapat dikembangkan dari skala kecil ke skala besar.

5. KESIMPULAN

Berdasarkan penelitian dapat disimpulkan telah dibangun sistem jaringan komputer dengan *dynamic*

routing BGP. Hasil pengujian dengan metode *comparison testing* menunjukkan kinerja sistem jaringan dengan menggunakan BGP lebih baik dibandingkan tanpa BGP. Perbandingan parameter rata-rata *latency* diperoleh nilai 0% (hampir tanpa *latency*) artinya kecepatan akses lebih cepat dibandingkan tanpa BGP; parameter *traceroute* (kontel lokal) 50% lebih baik dibandingkan tanpa BGP, namun untuk *traceroute* (kontel non lokal) memiliki nilai presentase yang sama hal ini dikarenakan seluruh *prefix* non lokal hanya didapatkan dari *port backbone* lama.

6. DAFTAR PUSTAKA

- [1] S. Iwan, "Cisco CCNA dan Jaringan Komputer," Bandung Inform. Bandung, pp. 244–248, 2010.
- [2] H. Wijaya, "Belajar Sendiri Cisco Router." Jakarta: PT Elex Media Komputindo, 2001.
- [3] S. S. Kusumawardani and I. Nabella, "SIMULASI PROSES PEMILIHAN JALUR DALAM BORDER GATEWAY PROTOCOL (BGP) BERDASARKAN ATRIBUT WEIGHT DAN LOCALPREFERENCE," Universitas Gadjah Mada, 2014.
- [4] A. F. Rochim, Y. Christiyono, and I. G. P. Yasa, "DESAIN DAN SIMULASI INTERNAL BORDER GATEWAY PROTOCOL (IBGP) MENGGUNAKAN GRAPHICAL NETWORK SIMULATOR (STUDI KASUS PADA JARINGAN UNIVERSITAS DIPONEGORO)," *Transmisi*, vol. 16, no. 1, pp. 20–25, 2014.
- [5] M. Caesar and J. Rexford, "BGP routing policies in ISP networks," *IEEE Netw.*, vol. 19, no. 6, pp. 5–11, 2005.
- [6] G. Huston, "Analyzing the Internet's BGP routing table," *Internet Protoc. J.*, vol. 4, no. 1, pp. 2–15, 2001.
- [7] Sukardi, "Metodologi Penelitian Pendidikan Kompetensi dan Praktiknya." PT Bumi Aksara, 2003.
- [8] R. Kuhn, K. Sriram, and D. Montgomery, "Border gateway protocol security," *NIST Spec. Publ.*, vol. 800, p. 54, 2007.
- [9] E. Chen, "Route Refresh Capability for BGP-4," 2000. [Online]. Available: <https://tools.ietf.org/html/rfc2918>. [Accessed: 09-Jan-2017].
- [10] A. B. Ali, M. Tabassum, and K. Mathew, "A comparative study of IGP and EGP routing protocols, performance evaluation along load balancing and redundancy across different AS," in *Proceedings of the International Multi-Conference of Engineers and Computer Scientists*, 2016, vol. 2.
- [11] "Latency Definition," The Linfo website, 2017. [Online]. Available: <http://www.linfo.org/latency>.
- [12] Traceroute, 2017. [Online], Available: <https://www.rumahweb.com/journal/traceroute-tracert.htm>.
- [13] K. Weitz, D. Woos, E. Torlak, M. D. Ernst, A. Krishnamurthy, and Z. Tatlock, "Scalable verification of border gateway protocol configurations with an SMT solver," *ACM SIGPLAN Not.*, vol. 51, no. 10, pp. 765–780, 2016.

Pelacak Orang Hilang Menggunakan Sepatu dengan Sistem GPS dan GSM

Mayang Rizqi Ambagapuri*, Feby Nurkhalih Sulistya Putra, Mifta Thahira, Umi Fadlilah

Program Studi Teknik Elektro
Universitas Muhammadiyah Surakarta
Surakarta

*d400160155@student.ums.ac.id

Abstrak- Kasus orang hilang merupakan salah satu kasus yang dalam sistem pencariannya membutuhkan petunjuk yang akurat. Salah satu metode pencarian yang akurat adalah dengan menggunakan teknologi GPS akan memberitahu di mana keberadaan lokasi seseorang dengan tepat. Artikel ini mendeskripsikan tentang alat pelacak dengan menggunakan sistem GPS dan modul GSM yang diterapkan pada sepatu, tujuannya adalah sebagai sistem pelacakan orang hilang dengan memberikan titik lokasi yang akurat. Perancangan dari alat ini meliputi studi literatur, pembuatan program pada rangkaian sistem mikrokontroler dengan menggunakan Arduino, modul GSM sim800 sebagai media untuk mengirim dan menerima SMS dari pengguna, modul GPS *tracking* sebagai pendeteksi titik koordinat keberadaan orang yang hendak dilacak, membuat desain untuk peletakkan komponen pada sepatu, dan metode terakhir yaitu uji coba alat. Hasil dari penelitian adalah sepatu dapat dilacak ketika alat yang ada di dalam sol sepatu sudah dalam keadaan *on/aktif*. Sepatu dapat dilacak dengan menggunakan SMS. Sepatu dapat dilacak menggunakan sms, dengan mengirim sms sesuai dengan kode yang sudah diprogram pada alat, jika kode sms yang dikirim benar maka alat akan membalas berupa titik koordinat dari pengguna sepatu.

Kata Kunci: alat pelacak, *arduino*, orang hilang, sepatu

1. Pendahuluan

Kasus orang hilang di antaranya disebabkan karena faktor bencana alam, pembunuhan, maupun penculikan termasuk penculikan anak dengan berbagai motif misalnya motif ekonomi [1]. Pola perdagangan anak di kota besar terdiri atas berbagai tipe yang memiliki karakteristik beragam baik dari sisi daerah asal, daerah transit, tujuan, faktor penarik dan pendorong, proses rekrutmen dan kondisi korban [2]. Menurut data dari Komnas Anak, kasus penculikan anak semakin meningkat setiap tahunnya. Pada tahun 2014 terdapat 51 kasus penculikan anak 6 di antaranya penculikan bayi dan di tahun 2015 terdapat 87 kasus penculikan anak. Pada tahun 2016 jumlah kasus penculikan anak menjadi 112 kasus kemudian pada tahun 2017 dihitung dari Januari hingga Maret terdapat 23 kasus penculikan anak [3].

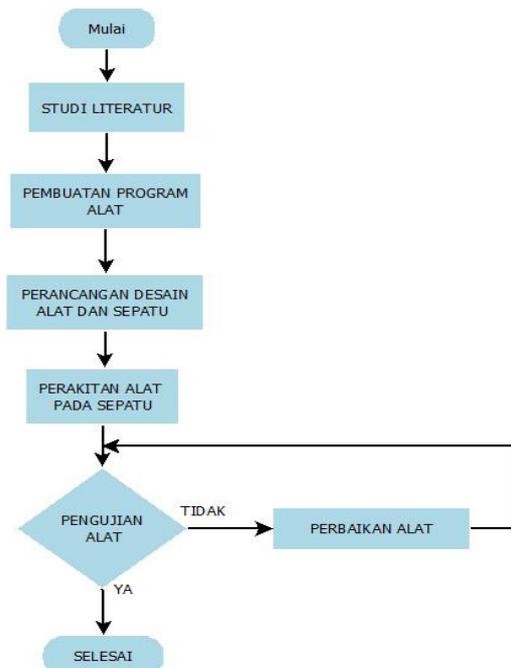
Banyak metode yang dapat dilakukan untuk melakukan pencarian seara akurat, misalnya dengan menggunakan teknologi GPS yang dapat memantau dan mengetahui di mana keberadaan atau lokasi seseorang. Metode penentuan posisi adalah mekanisme bagaimana memantau keberadaan objek bergerak di muka bumi, posisi objek ini didapatkan dari perpotongan *longitude* (garis bujur) dan *latitude* (garis lintang) [4]. Untuk mendapatkan titik lokasi koordinat diperoleh dengan sistem SMS yang dikirimkan ke modul GSM. Teknologi yang memanfaatkan GPS dan GSM sebelumnya telah dibuat oleh [5] dan [6]. Karya yang dibuat oleh [5] mengamati sistem keamanan pada sepeda motor untuk pencegahan dan pelacakan pencurian sepeda motor. Karya yang dibuat

oleh [6] GPS dan GSM dimanfaatkan untuk pemanggilan ambulans secara cepat pada saat terjadi kecelakaan. Sistem pelacakan sebelumnya juga sudah diterapkan oleh [7] yang mengamati pengembangan sistem pelacakan pada kendaraan yang berbasis GPS dan GSM dengan aplikasi bersistem operasi android, koordinat akan dikirimkan melalui SMS, kemudian koordinat dimasukkan melalui tampilan *maps* yang ada pada android. Ketiga penelitian terakhir ini mengindikasikan bahwa sistem pelacakan dengan menggunakan GPS dan GSM merupakan metode pencarian yang efisien dan akurat.

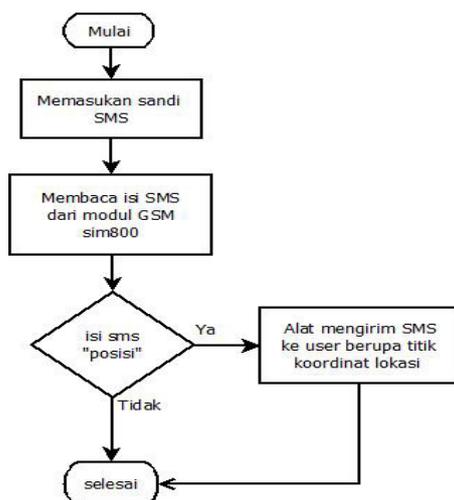
Berdasarkan uraian dua paragraf di atas penulis mengupayakan pemecahan masalah menemukan orang hilang dengan mengembangkan alat pelacakan menggunakan modul GPS dan modul GSM. Alat ini dipasang pada sepatu sehingga orang yang menggunakan sepatu ini dapat dilacak keberadaannya melalui pengiriman SMS berisi data posisi sepatu yang digunakan orang yang dilacak.

2. Metode

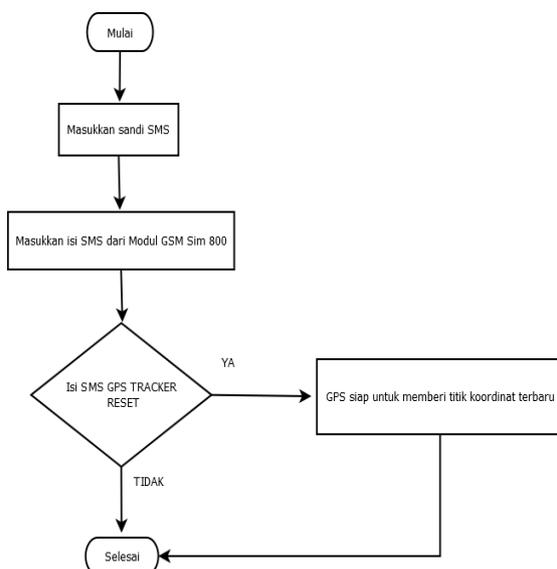
Metode penelitian dilakukan melalui eksperimen dan observasi. Langkah yang dilakukan adalah pembuatan alat, pengujian dan pengamatan. Tahapannya berupa perancangan program pada sistem mikrokontroler Arduino untuk menjalankan sistem GPS dan GSM sim800, membuat desain untuk meletakkan alat pada sol sepatu, perakitan alat pada sepatu, kemudian pengujian alat untuk mendapatkan hasil (lihat Gambar 1).



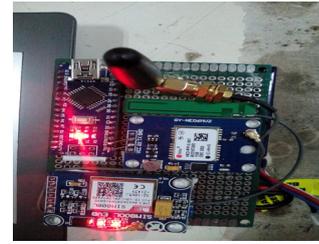
Gambar 1. Diagram alir pelaksanaan



Gambar 2. Diagram alur meminta titik koordinat lokasi



Gambar 3. Diagram alur untuk set ulang GPS



Gambar 4. Tampilan alat untuk sepatu



Gambar 5. Tampilan indikator led GPS menyala

a. Alur Kerja Alat

Pada penelitian ini terdapat dua diagram alur yang masing-masing menunjukkan sistem kerja modul GPS yang ada pada alat. Gambar 2 menunjukkan diagram alur yang menjelaskan kerja GPS untuk mengirimkan titik koordinat ke *user* melalui SMS. Ketika *user* ingin mencari keberadaan orang yang hilang alat sudah harus dalam keadaan saklar *on*/aktif, yang ditandai dengan menyalnya indikator led pada rangkaian Arduino. Pada saat itu, alat akan mengirimkan SMS "HUTRACK ONLINE". Selanjutnya *user* dapat memasukkan sandi SMS dengan kode "posisi" maka SMS tersebut akan masuk dan diproses oleh modul GSM sim800 untuk diinisialisasi apakah kode tersebut benar atau tidak. Selanjutnya jika benar maka alat akan membalas dengan mengirimkan titik koordinat lokasi dari orang yang memakai sepatu pelacak ini. Jika SMS yang dikirim bukan berupa sandi yang sesuai maka alat tidak akan membalas.

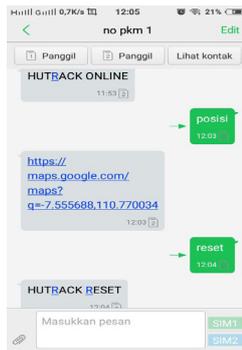
Diagram alur yang kedua pada Gambar 3 menunjukkan proses untuk mengatur ulang GPS. Tujuan untuk mengatur ulang GPS ini agar GPS tidak mengirim kembali di titik lokasi sebelumnya, tetapi mengirimkan lokasi alat sesuai keberadaan terbaru. *User* harus mengirimkan SMS dengan kode "RESET" kemudian SMS ini akan diinisialisasi oleh GSM sim800 dan jika alat sudah membalas SMS "HUTRACK RESET" dan "HUTRACK ONLINE" maka GPS siap untuk mencari titik koordinat terbaru.

3. Hasil

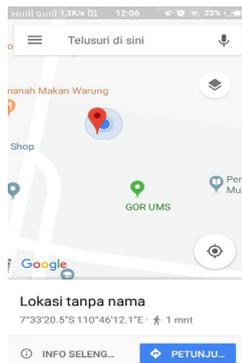
a. Pengembangan Alat

Hasil dari perancangan alat sepatu pelacak ditampilkan pada Gambar 4-10.

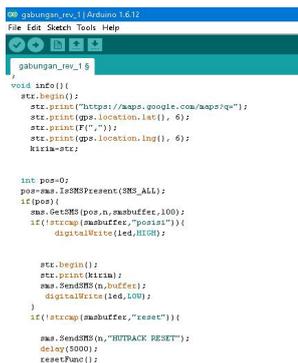
Gambar 4 menampilkan alat untuk dimasukkan dalam sol sepatu. Pengujian menunjukkan bahwa ketika saklar yang ada pada sepatu dihidupkan maka indikator led pada Arduino akan menyala. Indikator led pada GSM dan GPS menyala ketika GSM dan GPS sudah menangkap sinyal. Pada Gambar 5, Indikator led modul GPS sudah menyala maka GPS sudah siap mencari titik koordinat.



Gambar 6. Pengiriman SMS.



Gambar 7. Tampilan pada google maps



Gambar 8. Program untuk kode meminta link titik koordinat yang dikirim lewat SMS



Gambar 9. Pemasangan alat pada sol sepatu



Gambar 10. Alat pelacak dalam sol sepatu

Apabila semua indikator led sudah menyala maka alat akan mengirimkan SMS “HUTRACK ONLINE” ke nomor *user* dan alat siap untuk mencari titik koordinatnya. Gambar 6 memperlihatkan cara pengiriman SMS untuk mencari titik koordinat letak sepatu dengan format “posisi”. Hasilnya dapat dilihat pada Gambar 6 bahwa alat akan membalas dengan memberikan titik koordinat. Titik koordinat yang dikirim ke *user* berupa link yang dapat dibuka melalui Google Maps pada *smartphone*. Data link tersebut awalnya merupakan data *latitude* dan *longitude* yang di-*parsed*, sehingga menjadi satu kesatuan data berupa link yang dapat dibuka (lihat Gambar 7).

b. Pengujian Alat

Pengujian alat dilakukan dua tahap, pengujian sistem kerja alat dan pengujian untuk mendeteksi lokasi alat. Pengujian pertama, yaitu menguji kerja alat dari awal sampai akhir, sesuai dengan yang diharapkan. Pengujian yang kedua mendeteksi lokasi alat untuk mendapatkan data berupa titik koordinat. Pengujian sistem kerja alat ditampilkan pada Tabel 1 terdapat empat tahap dalam pengujian kerja, dari menyalakan saklar pada sepatu hingga mereset GPS.

Pengujian kedua adalah deteksi lokasi alat yang dilakukan di dua tempat berbeda. Pada pengamatan pertama, sepatu digunakan pemakai dalam keadaan diam di Laboratorium Robot Research Kampus 2 UMS. Pada pengamatan kedua, sepatu dipakai dalam keadaan berjalan di sebuah lokasi di Banyudono, Boyolali. Data yang diambil berupa titik koordinat *latitude* dan *longitude*. Tabel 2 menampilkan hasil pengujian dari tempat pertama yaitu sebanyak 5 kali percobaan. Hasilnya adalah 5 titik koordinat berbeda namun saling berdekatan.

Data pengujian tempat kedua diambil saat pemakai sepatu dalam keadaan berjalan dan berada di luar ruangan. Lokasinya yaitu di Perumahan Ngaru-Aru Indah, Banyudono, Boyolali. Data yang didapatkan dari 5 kali percobaan menghasilkan data *latitude* dan *longitude* yang berbeda, akan tetapi jika dibandingkan dengan lokasi pertama, lokasi kedua menghasilkan data yang tidak terlalu jauh hasilnya pada data *longitude*-nya, data longitude ke-1 dengan ke-2 menghasilkan data yang sama, begitupun data ke-4 dengan ke-5 menghasilkan data yang sama pula. Titik koordinat pengujian mengalami selisih yang berbeda karena pemakai sepatu dalam keadaan berjalan.

Tabel 1. Pengujian skenario

Skenario Pengujian	Hasil yang diharapkan	Pengamatan	Keterangan
Mengaktifkan saklar	Indikator led pada Arduino menyala sebagai tanda alat sudah aktif	Led merah Arduino menyala	Berhasil
Menunggu SMS "HUTRACK ONLINE"	Indikator led pada GSM dan GPS menyala tanda telah mendapatkan sinyal, dan alat mengirimkan SMS	Led merah modul GSM dan GPS menyala dalam waktu 3 detik setelah saklar diakifikan dan mengirim pesan penanda alat telah online.	Berhasil
User mengirim sms dengan kode "posisi"	Alat membalas dengan mengirimkan SMS titik lokasi orang yang memakai sepatu	Alat mengeksekusi perintah dan mengirimkan data lokasi berupa data <i>longitude</i> dan <i>latitude</i> yang sudah diparsing menjadi <i>link</i> yang dapat dibuka melalui <i>google maps</i>	Berhasil
Merest alat untuk mendapat titik koordinat baru dengan mengirim sms dengan kode "HUTRACK RESET"	Alat membalas dengan mengirim SMS penanda	Alat mengeksekusi perintah dan mengirimkan SMS "HUTRACK RESET" dan selanjutnya mengirim SMS "HUTRACK ONLINE" kemudian alat dapat mengirimkan titik lokasi terbaru.	Berhasil

Tabel 2. Data pengujian titik koordinat lokasi pertama

Lokasi	Titik Koordinat	
	<i>latitude</i>	<i>longitude</i>
Kampus 2 UMS	-7.555518	110.769790
Jl Ahmad Yani Tromol Pos 1	-7.555500	110.769851
Pabelan, Kartasura, Surakarta	-7.555578	110.769966
	-7.555600	110.769950
	-7.555602	110.769981

Tabel 3. Data pengujian titik koordinat lokasi kedua

Lokasi	Titik Koordinat	
	<i>latitude</i>	<i>longitude</i>
	-7.546100	110.677597
	-7.546132	110.677597
Banyudono, Boyolali	-7.546083	110.677574
	-7.546098	110.677558
	-7.546092	110.677558

Berdasarkan pengujian di dua lokasi menghasilkan data dengan tingkat keakuratan berbeda. Hal ini dapat diamati bahwa ruang juga bisa mempengaruhi hasil. Pada saat alat berada di luar ruangan akan menghasilkan data yang lebih akurat dibandingkan ketika berada dalam ruangan, ini disebabkan pencarian posisi melalui GPS dalam ruangan kurang efektif karena sinyal satelit yang tertutup oleh ruangan [8].

4. Diskusi

Alat yang telah dibuat dapat digunakan sebagai pelacak untuk orang hilang menggunakan sistem GPS yang menghasilkan data mendekati akurat sesuai dengan pengujian di atas. Berdasarkan hasil pengujian titik koordinat yang dikirim ke *user* berupa *link* yang awalnya adalah data *latitude* dan *longitude*. *Latitude* dan *longitude* didapatkan dari sistem Arduino yang berperan sebagai mikrokontroler yang telah menerima data dari modul

GPS yang kemudian diekstrak menjadi angka *latitude* dan *longitude* [9]. Angka *latitude* dan *longitude* di-*parse* sehingga menjadi satu kesatuan data berupa alamat *link* yang dapat dibuka pada Google Maps.

Pengujian yang telah dilakukan menghasilkan titik koordinat yang mendekati akurat. Sinyal gelombang mikro GPS berjalan dengan kecepatan cahaya, sehingga GPS tidak bisa menentukan posisi lebih besar dari akurasi 3 meter [10]. Tingkat keakuratan GPS juga dipengaruhi oleh ruang. Pencarian GPS jika di dalam ruangan kurang efektif karena sinyal satelit tertutup oleh ruangan [8].

Alat pelacakan sebelumnya sudah dibuat oleh [11] bernama *child tracker*, sebuah aplikasi pada *smartphone* android yang digunakan untuk mengetahui keberadaan anak yang membantu memberikan lokasi, dan menghubungi orang tua. Berbeda dengan penelitian kami yang memasang perangkat pada sepatu, alat *child tracker* memanfaatkan GPS yang berada pada *smartphone* yang dihubungkan melalui aplikasi. Kelemahan dari aplikasi

child tracker ini adalah *smartphone* harus selalu terhubung dengan internet, sehingga ketika tidak mendapat sinyal internet maka proses pengiriman perintah pada aplikasi tidak berjalan [11]. Alat pelacak orang hilang yang dibuat oleh peneliti tidak membutuhkan koneksi internet. Modul GPS hanya membutuhkan sinyal satelit yang tersedia di seluruh area di bumi selama modul GPS mendapatkan sinyal satelit.

Selain itu telah dikembangkan pula alat pelacakan kendaraan yang memanfaatkan modul GPS dan GSM. Alat tersebut digunakan untuk pencegahan pencurian kendaraan sebagai kelengkapan tambahan selain kunci pengaman. Alat mampu mendeteksi keberadaan kendaraan dalam jangkauan delapan meter dari titik koordinat yang terdeteksi baik dalam keadaan diam maupun bergerak [12].

5. Kesimpulan

Berdasarkan pengujian yang telah dilakukan dapat disimpulkan bahwa alat dapat bekerja dengan baik sesuai dengan yang diharapkan serta memiliki 2 fitur yaitu SMS dan GPS. Alat ini dapat merespons ketika ada perintah yang masuk. Dengan menggunakan sistem GPS dan GSM yang dikontrol oleh Arduino maka alat dapat dipantau dari jarak dekat maupun jauh.

6. Persantunan

Penulis mengucapkan terima kasih kepada Kementerian Ristekdikti yang telah memberikan dana sehingga penelitian ini bisa berjalan dengan baik.

7. Daftar Pustaka

- [1] E. Srihadi, "Laporan Utama: Fenomena Penculikan Anak," *Update Indonesia*, Jakarta, pp. 2–5, 2010.
- [2] R. R. Ambarsari, F. A. Andiyansyah, and A. A. Soewandi, "Kajian Yuridis Tentang Perlindungan Hukum Terhadap Perempuan dan Anak Korban Human Trafficking," *J. Panor. Huk.*, vol. 1, no. 1, pp. 57–73, 2016.
- [3] A. Setiawan and A. Sadat, "Kasus Penculikan Anak Setiap Tahun Meningkat," *Viva News*, 2017. [Online]. Available: <http://www.viva.co.id/berita/nasional/897701>.
- [4] Hanafi, "Aplikasi Pemantauan Keberadaan Lokasi dan Kecepatan pada Kendaraan dengan Menggunakan Teknologi Mobile Data dan GPS dengan Digitalisasi Peta," *J. Teknol.*, vol. 8, no. 2, 2015.
- [5] D. Pratama, D. A. Hakim, Y. Prasetya, N. R. Febriandika, M. Trijati, and U. Fadlilah, "Rancang Bangun Alat dan Aplikasi untuk Para Penyandang Tunanetra Berbasis Smartphone Android," *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 2, no. 1, pp. 14–19, 2016.
- [6] S. Sonika, D. K. Sathiyasekar, and S. Jaishree, "Intelligent accident identification system using GPS, GSM modem," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 3, no. 2, 2014.
- [7] H. H. Irfanhady, "Security car system based GPS and SMS," Telkom University, 2015.
- [8] F. M. Asmawi, "Rancang Bangun Sistem Navigasi Indoor Berbasis Integrasi Symbolic Location Model Dan Wi-Fi Based Positioning System Untuk Studi Kasus Pada Gedung Bertingkat," Institut Teknologi Sepuluh Nopember, 2017.
- [9] R. Karthik, S. Ranjith, S. Shreyas, and P. Kumar, "Automatic Border Alert System for Fishermen using GPS and GSM techniques," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 7, no. 1, pp. 84–89, 2017.
- [10] Anil, "Menegal Lebih Dalam Mengenai GPS, Navigasi yang Memudahkan Perjalanan Anda," 2018. [Online]. Available: <http://www.sfdcs.org/teknologi/menegal-lebih-dalam-mengenai-gps-navigasi-yang-memudahkan-perjalanan-anda/>. [Accessed: 29-Jun-2018].
- [11] A. Juansyah, "Pembangunan aplikasi child tracker berbasis assisted-global positioning system (a-gps) dengan platform android," *J. Ilm. Komput. dan Inform. Ed.*, vol. 1, 2015.
- [12] D. Pratama, E. D. Febrianto, D. A. Hakim, T. Mulyadi, R. W. Halfiah, and U. Fadlilah, "Sistem Keamanan Ganda pada Sepeda Motor untuk Pencegahan Pencurian dengan SMARTY (Smart Security)," *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 3, no. 1, pp. 31–37, 2017.

Aplikasi Sistem Pakar Pendeteksi Penyakit Tuberkulosis Berbasis Android

Eko Didik Widiyanto*, Yuni Waz Zaituun, Ike Pertiwi Windasari

Departemen Teknik Sistem Komputer, Fakultas Teknik, Universitas Diponegoro
Jl. Prof. Soedarto, SH, Kampus Undip Tembalang, Semarang, Indonesia 50275

*didik@live.undip.ac.id

Abstrak-Penelitian ini mengembangkan aplikasi sistem pakar secara khusus untuk mendeteksi dini penyakit tuberkulosis paru, tuberkulosis tulang dan tuberkulosis kelenjar. Aplikasi dikembangkan untuk berjalan di perangkat Android. Proses diagnosis menggunakan metode forward chaining dari 18 gejala ketiga penyakit tersebut, baik gejala wajib maupun gejala tidak wajib. Basis data menggunakan SQLite. Untuk menghitung jumlah gejala yang dipilih oleh pengguna terhadap jumlah gejala penyakit tersebut, digunakan probabilitas klasik. Secara fungsional, sistem pakar telah dapat berjalan di perangkat Android untuk mendeteksi dini penyakit tuberkulosis paru, tulang dan kelenjar serta telah divalidasi hasil deteksinya oleh pakar secara langsung.

Kata Kunci: tuberkulosis, sistem pakar, aplikasi Android, *forward chaining*, teori probabilitas klasik

1. Pendahuluan

Tuberkulosis (TBC) merupakan penyakit menular yang umum dan dalam beberapa kasus bersifat mematikan. Penyakit ini disebabkan oleh berbagai *strain mikobakteria*, umumnya *mycobacterium tuberculosis* (disingkat “MTb” atau “MTbc”). Tuberkulosis biasanya menyerang paru-paru, namun juga bisa berdampak pada bagian tubuh lainnya. Tuberkulosis menyebar melalui udara ketika seseorang dengan infeksi tuberkulosis aktif batuk, bersin atau menyebarkan butiran ludah mereka melalui udara [1].

World Health Organization (WHO) menyatakan bahwa penyakit tuberkulosis paru saat ini telah menjadi ancaman global karena hampir sepertiga penduduk dunia telah terinfeksi [2]. Sebanyak 95% kasus tuberkulosis dan 98% kematian akibat tuberkulosis di dunia, terjadi pada negara-negara berkembang. Tuberkulosis merupakan penyebab kematian nomor satu di antara penyakit menular dan merupakan peringkat ketiga dari 10 penyakit pembunuh tertinggi di Indonesia yang menyebabkan 100.000 kematian setiap tahunnya [3]. Meningkatnya jumlah penderita tuberkulosis dipengaruhi oleh banyaknya rakyat miskin dengan pola hidup yang tidak sehat, lingkungan yang tidak bersih dan kurangnya informasi tentang penyakit tersebut beserta gejala dan penyebabnya yang akan membuat proses penanganan menjadi lambat. Proses penanganan yang lambat dan tidak tepat akan membuat penyakit semakin parah dan berakibat fatal.

Saat ini teknologi telah banyak dimanfaatkan dalam bidang medis. Salah satunya adalah sistem pakar. Sistem pakar adalah program yang dirancang untuk menirukan keahlian seorang pakar dalam menjawab pertanyaan dan menyelesaikan suatu permasalahan pada berbagai bidang. Dengan sistem pakar, orang awam dapat menyelesaikan

masalah dengan informasi yang akurat yang sebenarnya hanya dapat diselesaikan oleh ahli atau pakarnya, seperti untuk deteksi dini meningitis dan gastroenteritis serta penentuan kebutuhan gizi individu dalam [4]-[6].

Pengembangan sistem pakar untuk deteksi dini tuberkulosis juga telah banyak dilakukan [7]-[11]. Platform aplikasi beragam, yaitu berbasis web [7]-[8] menggunakan PHP-MySQL dan *desktop* [9]-[11] menggunakan Delphi dan Visual Basic. Aplikasi dapat mendeteksi secara khusus penyakit tuberkulosis [8]-[10] dan tuberkulosis ditambah penyakit lain [7], [11]. Algoritma sistem pakar yang digunakan adalah antara lain *forward chaining* seperti dalam [9] dan [11], *certainty factor* (CF) dalam [7], [10], dan fuzzy Sugeno dalam [9].

Penelitian ini bertujuan untuk mengembangkan aplikasi sistem pakar untuk deteksi dini penyakit tuberkulosis menggunakan metode *forward chaining* seperti dalam [9], [11]. Berbeda dengan Arsyad [9] serta Mulyani dan Restianie [11] yang mengembangkannya berbasis *desktop*, aplikasi ini dikembangkan untuk dijalankan di perangkat berbasis Android seperti dalam [4]-[5] yang mengembangkannya untuk mendeteksi dini meningitis dan gastroenteritis. Secara khusus, penyakit TBC yang hendak dideteksi oleh aplikasi yang dikembangkan dalam penelitian ini adalah tuberkulosis paru, tuberkulosis tulang, dan tuberkulosis kelenjar.

2. Metode

Tahapan pengembangan aplikasi deteksi dini tuberkulosis ini adalah meliputi koleksi pengetahuan, perancangan sistem, dan pengujian. Lingkup jenis penyakit yang dideteksi adalah tuberkulosis paru, tuberkulosis tulang, dan tuberkulosis kelenjar.

id	nama	ya	tidak	mulai	selesai
Fi...	Filter	F...	Fil...	F...	Fil...
G001	Demam	G002	B	Y	T
G002	Lemah, Letih, Lesu, tidak nafsu makan	G003	B	T	T
G003	Penurunan berat badan	G004	G004	T	T
G004	Keringat dingin waktu malam hari	G005	B	T	T
G005	Batuk berdahak yang berlangsung lebih...	G006	B	T	T
G006	Batuk yang mengeluarkan darah	G007	G015	T	T
G007	Terasa sesak nafas dan nyeri pada dada	P001	P001	T	T
G008	Terasa nyeri pada sendi atau tulang	G009	G007	T	T
G009	Pegal-pegal dibarengi rasa capek	G010	G010	T	T
G010	Bengkak pada bagian tubuh yang nyeri	G011	G012	T	T
G011	Kulit yang bengkak berwarna merah ke...	G012	G012	T	T
G012	Kulit di atas tempat yang merasa nyeri ...	G013	G013	T	T
G013	Kelemahan pada otot	G014	G014	T	T
G014	Keterbatasan untuk melakukan pergera...	P002	P002	T	T
G015	Benjolan tanpa rasa sakit yang berlang...	G016	G008	T	T
G016	Benjolan yang terdapat pada daerah ke...	G017	G017	T	T
G017	Benjolan yang memiliki cairan kental b...	G018	G018	T	T
G018	Benjolan yang terus tumbuh baik ukura...	P003	P003	T	T
P001	Tuberkulosis Paru	0	0	T	Y
P002	Tuberkulosis Tulang	0	0	T	Y
P003	Tuberkulosis Kelenjar	0	0	T	Y

Gambar 1. Struktur Tabel Gejala

kode_penyakit	nama_penyakit	detail	penanganan
Filter	Filter	Filter	Filter
P003	Tuberkulosis Kelenjar	Berbeda dengan jenis TBC paru-paru dan TBC tulang, jenis penyakit TBC kelenjar akan menyerang area kelenjar getah bening pada tubuh dan bagian tubuh seperti pada leher, ketiak atau lipatan paha. Meskipun secara umum gejala TB kelenjar sama seperti batuk dan berkerasat dingin, tipe Tuberkulosis kelenjar ini akan menimbulkan benjolan pada daerah-daerah yang rawan terkena TBC. Benjolan awalnya berukuran kecil. Namun, jika dibiarkan akan terus membesar	1. Melakukan pemeriksaan diri ke dokter jika Anda mengalami gejala-gejala TBC Kelenjar 2. Menjaga kebersihan diri dan lingkungan sekitar 3. Hidup sehat dengan berolahraga secara teratur dan Makan makanan yang bergizi 4. Pengobatan yang diberikan...
P002	Tuberkulosis Tulang	Tuberkulosis tulang bisa menyerang nyaris seluruh tulang namun yang sangat kerap berlangsung yaitu tbc pada tulang belakang, pinggul, lutut, kaki, siku, tangan serta bahu, rahang bawah serta sendi temporomandibular yaitu tempat yang sangat jarang terjangkit oleh kuman tbc	1. Melakukan pemeriksaan diri ke dokter jika Anda mengalami gejala-gejala TBC Tulang 2. Hidup sehat dengan berolahraga secara teratur dan Makan makanan yang bergizi ...
P001	Tuberkulosis Paru	Bila infeksi Tuberkulosis yang timbul menjadi aktif, sekitar 90 persennya selalu melibatkan paru-paru. Gejala-gejalanya antara lain berupa nyeri dada dan batuk berdahak yang berkepanjangan. Sekitar 25 persen penderita tidak menunjukkan gejala apapun (yang demikian disebut asimtomatik). Kadangkala, penderita mengalami sedikit batuk darah. Dalam kasus-kasus ter...	1. Melakukan pemeriksaan diri ke dokter jika Anda mengalami gejala-gejala TBC 2. Meminum obat yang telah diberikan oleh dokter secara teratur dan jangan menghentikan pemakaian obat...

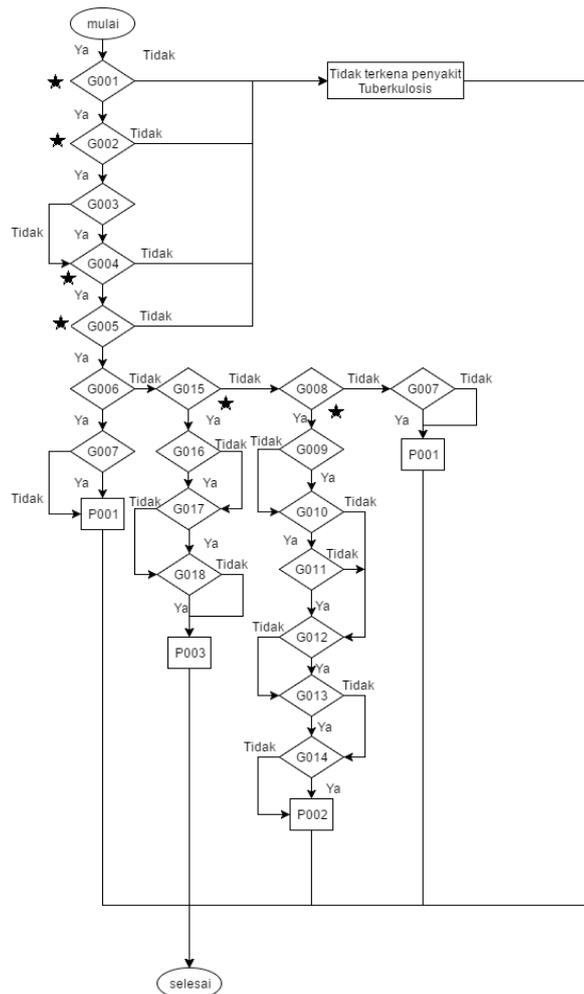
Gambar 2. Struktur Tabel Penyakit

Tabel 1. Kode Gejala dan Penyakit

Kode	Nama Gejala dan Penyakit
G001	Demam
G002	Lemah, letih, lesu, tidak nafsu makan
G003	Penurunan berat badan
G004	Keringat dingin waktu malam hari
G005	Batuk berdahak yang berlangsung lebih dari 14 hari
G006	Batuk yang mengeluarkan darah
G007	Terasa sesak nafas dan nyeri dada
G008	Terasa nyeri pada sendi atau tulang
G009	Pegal-pegal dibarengi rasa capek
G010	Bengkak pada bagian tubuh yang nyeri
G011	Kulit yang bengkak berwarna merah kebiruan
G012	Kulit di atas tempat yang merasa nyeri terkadang terasa panas dan terkadang terasa dingin
G013	Kelemahan pada otot
G014	Keterbatasan untuk melakukan pergerakan
G015	Benjolan tanpa rasa sakit yang berlangsung lama
G016	Benjolan yang terdapat pada daerah kelenjar getah bening, misal di leher atau ketiak
G017	Benjolan yang memiliki cairan kental berupa nanah
G018	Benjolan yang terus tumbuh baik ukuran maupun jumlahnya
P001	Tuberkulosis Paru
P002	Tuberkulosis Tulang
P003	Tuberkulosis Kelenjar

Tabel 2. Tabel keputusan sistem pakar

Kode Gejala	Kode Penyakit		
	P001	P002	P003
G001	*	*	*
G002	*	*	*
G003	v	v	v
G004	*	*	*
G005	*	*	*
G006	v		
G007	v		
G008		*	
G009		v	
G010		v	
G011		v	
G012		v	
G013		v	
G014		v	
G015			*
G016			v
G017			v
G018			v



Gambar 3. Diagram alir program

Tahap koleksi pengetahuan dilakukan dengan menggali pengetahuan dari pakar tentang penyakit tuberkulosis, jenis, dan gejalanya. Pengetahuan ini digunakan sebagai basis pengetahuan pakar untuk aturan produksi sistem dalam tahap perancangan dan implementasi sistem pakar. Tahap ini merancang desain tampilan dan antarmuka sistem, desain basis data sistem, serta cara kerja sistem. Hasil dari perancangan diimplementasikan ke dalam aplikasi sistem pakar. Aplikasi ini dibuat dengan *software* Android Studio dan basis data SQLite. Tahap pengujian sistem pakar dilakukan untuk mengevaluasi sistem secara fungsional.

Pembuatan basis data dilakukan dengan menggunakan basis data SQLite yang diprogram langsung dari kelas basis data. Dalam tahap ini terdapat dua buah *query* untuk membentuk dua buah tabel, yaitu tabel gejala dan tabel penyakit.

Tabel gejala berisi kolom id, nama, ya, tidak, mulai, dan selesai dengan id sebagai *primary key* yang ditunjukkan dalam Gambar 1. Kolom id berisi isian dengan kode gejala yang unik. Kolom nama berisi nama-nama gejala dari suatu penyakit. Kolom ya berisi id gejala dan penyakit di mana saat pengguna memilih jawaban ya. Kolom tidak berisi id gejala dan penyakit yang jika pengguna memilih jawaban tidak. Kolom mulai dan selesai berisi teks Y dan T, di mana jika pertanyaan pertama dimulai dan pertanyaan yang menyatakan telah selesai akan berisi Y. Terdapat 18 jumlah gejala yang dibutuhkan untuk mendefinisikan gejala-gejala dari tiap penyakit.

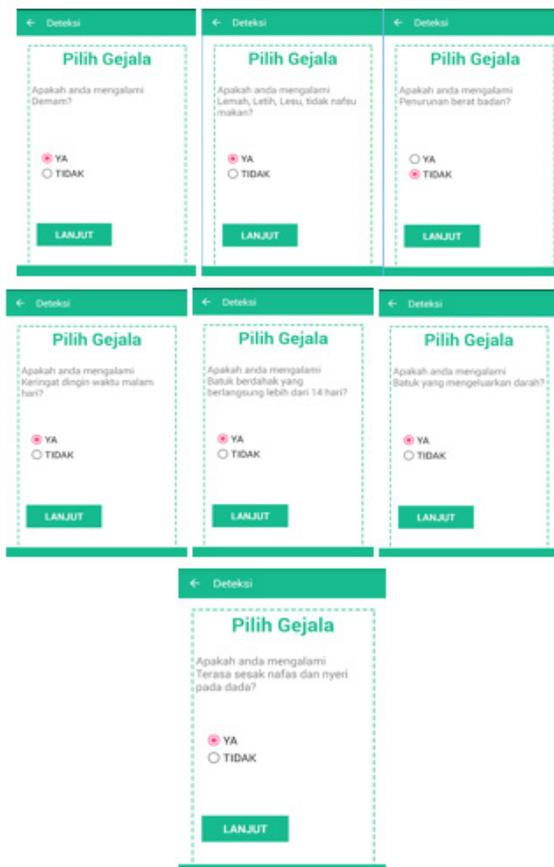
Tabel penyakit berisi kolom kode_ penyakit sebagai *primary key*, nama_ penyakit, detail, dan penanganan yang ditunjukkan dalam Gambar 2. Kolom kode diberi format isian P00+ nomor urutan penyakit. Kolom nama_ penyakit berisi nama dari masing-masing penyakit. Pada sistem pakar pendeteksi penyakit tuberkulosis ini, data penyakit hanya terdiri dari 3 jenis penyakit tuberkulosis yaitu tuberkulosis paru, tuberkulosis tulang, dan tuberkulosis kelenjar. Kolom detail berisi deskripsi dari masing-masing penyakit sedangkan kolom penanganan berisi upaya pencegahan dan penanganan dari tiap penyakit.

Dalam sistem pakar ini, penelusuran hasil deteksi menggunakan penelusuran runut maju. Alur penelusuran maju ditunjukkan oleh Gambar 3. Pada diagram alir ini disebutkan kode-kode gejala dan penyakit. Daftar kode gejala dan penyakit dinyatakan pada Tabel 1.

Proses deteksi pada masing-masing penyakit memiliki gejala yang wajib dipilih dan gejala yang tidak wajib dipilih. Pada diagram alir Gambar 3 terdapat tanda bintang yang menandakan gejala wajib dipilih. Gejala yang wajib dipilih adalah G001, G002, G004, dan G005. Jika salah satu dari gejala yang wajib dipilih tidak terpenuhi, maka sistem akan menyatakan bahwa penyakit yang diderita bukan termasuk penyakit tuberkulosis. Penyakit tuberkulosis tulang (P002) memiliki gejala khusus yaitu G008. Jika pengguna memilih gejala G008, maka akan mengarah ke gejala selanjutnya yang dimiliki oleh penyakit tuberkulosis tulang dan hasilnya akan dihitung nilai probabilitasnya. Penyakit tuberkulosis kelenjar (P003) juga memiliki gejala khusus yaitu G015. Jika pengguna memilih gejala G015, maka akan mengarah ke gejala selanjutnya yang dimiliki oleh penyakit tuberkulosis kelenjar dan akan dihitung nilai probabilitasnya. Jika kedua gejala khusus tersebut tidak dipilih pengguna maka gejala selanjutnya akan mengarah ke gejala yang dimiliki penyakit tuberkulosis paru.



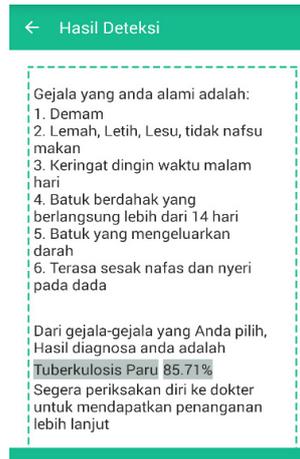
Gambar 4. Halaman Utama dan Deteksi Aplikasi



Gambar 5. Pertanyaan Gejala (Kasus)

Gejala yang dipilih memiliki aturan produksi yang diperoleh dari tabel keputusan. Keputusan tersebut berdasarkan dari gejala umum ke gejala khusus. Gejala yang sudah diberikan diklasifikasikan per penyakit yang mempunyai gejala yang sama. Hasil pengujian didapatkan dari tabel keputusan seperti dinyatakan pada Tabel 2. Tanda * menunjukkan gejala yang wajib dipilih, sedangkan tanda v menunjukkan gejala yang tidak wajib dipilih.

Teori probabilitas klasik digunakan dalam aplikasi untuk menunjukkan persentase gejala penyakit yang dipilih oleh pengguna dengan total gejala penyakit. Persentase tersebut dinyatakan dalam $P(E)$ sesuai Persamaan 1. Parameter a menyatakan gejala yang dipilih, sedangkan b menyatakan jumlah total gejala penyakit tersebut sesuai



Gambar 6. Hasil Deteksi Penyakit



Gambar 7. Hasil Pengujian Pakar

Tabel 3. Pengujian Fungsi Halaman Hasil Deteksi

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Menguji hasil deteksi	Memilih gejala-gejala yang dialami	Tampil gejala-gejala yang dipilih, hasil diagnosis, dan persentase kemungkinan penyakit	Berhasil
Menguji fungsi menu "Back"	Tekan tombol menu "Back"	Tampil halaman Awal sistem pakar	Berhasil

Tabel 4. Rencana pengujian fungsional sistem pakar

Item Pengujian	Detail Pengujian	Jenis Pengujian
Fungsi Halaman Utama	Pengujian fungsi "Mulai Deteksi"	Black Box
	Pengujian fungsi "Informasi Penyakit"	Black Box
Fungsi Halaman Mulai Deteksi	Pengujian fungsi "Tentang Aplikasi"	Black Box
	Pengujian masukan "Ya"	Black Box
	Pengujian masukan "Tidak"	Black Box
	Pengujian masukan "Lanjut"	Black Box
	Pengujian masukan "Back"	Black Box
Halaman Hasil Deteksi	Pengujian hasil deteksi penyakit	Black Box
	Pengujian fungsi "Back"	Black Box
Halaman Informasi Penyakit	Pengujian fungsi informasi penyakit	Black Box
	Pengujian fungsi "Back"	Black Box

Tabel 5. Pengujian fungsi halaman informasi penyakit

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Menguji fungsi halaman informasi penyakit	Klik tombol “ <i>tab view</i> ” pada nama penyakit	Menampilkan halaman informasi penyakit dengan 3 buah data penyakit yang berisi deskripsi, gejala, serta upaya pencegahan dan penanganan.	Berhasil
Menguji fungsi menu “ <i>Back</i> ”	Klik tombol menu “ <i>Back</i> ”	Tampil halaman Awal sistem pakar	Berhasil

Tabel 6. Pengujian fungsi halaman utama

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Menguji fungsi menu “ <i>Deteksi</i> ”	Tekan tombol menu “ <i>Mulai Deteksi</i> ”	Tampil halaman <i>Mulai Deteksi</i>	Berhasil
Menguji fungsi menu “ <i>Informasi Penyakit</i> ”	Tekan tombol menu “ <i>Informasi Penyakit</i> ”	Tampil halaman <i>Informasi Penyakit</i>	Berhasil
Menguji fungsi menu “ <i>Tentang Aplikasi</i> ”	Tekan tombol menu “ <i>Tentang Aplikasi</i> ”	Tampil halaman <i>Tentang Aplikasi</i>	Berhasil

Tabel 7. Pengujian fungsi halaman deteksi

Nama Pengujian	Bentuk Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Menguji fungsi masukan “ <i>Ya</i> ”	Tekan tombol masukan “ <i>Ya</i> ”	Memilih gejala penyakit yang ditampilkan	Berhasil
Menguji fungsi masukan “ <i>Tidak</i> ”	Tekan tombol masukan “ <i>Tidak</i> ”	Tidak memilih gejala penyakit yang ditampilkan	Berhasil
Menguji fungsi masukan “ <i>Lanjut</i> ”	Tekan tombol masukan “ <i>Lanjut</i> ”	Menampilkan gejala selanjutnya	Berhasil

Tabel 2. Persentase ini kemudian ditampilkan dalam aplikasi.

$$P(E) = \frac{a}{b} \times 100\% \quad (1)$$

3. Hasil dan Diskusi

Aplikasi pakar deteksi tuberkulosis ini berjalan di sistem Android. Tampilan awal dan halaman deteksi yang membutuhkan interaksi pengguna ditunjukkan dalam Gambar 4.

Halaman utama aplikasi terdapat 3 menu yaitu menu *Mulai Deteksi*, menu *Informasi Penyakit*, dan menu *Tentang Aplikasi*. Ketika pengguna mengklik menu *Mulai Deteksi* maka akan diarahkan ke halaman *Deteksi*. Pada halaman *Deteksi* berisi pertanyaan gejala dengan jawaban ya atau tidak. Ketika pengguna menekan tombol menu *Informasi Penyakit* maka diarahkan ke halaman *Informasi Penyakit* yang berisi data-data penyakit. Halaman *Informasi Penyakit* berisi informasi dari data-data penyakit yang ditampilkan dalam *tab view*. Isi dari data penyakit adalah deskripsi dari satu penyakit, gejala, dan upaya pencegahan dan penanganan. Ketika pengguna menekan tombol menu *Tentang Aplikasi* maka tampilan akan beralih ke halaman *Tentang Aplikasi* yang berisi penjelasan aplikasi dan data pengembang aplikasi.

Halaman *Deteksi* berisi pertanyaan gejala dimulai dari gejala umum kemudian mengarah ke gejala khusus. Pengguna memilih gejala dengan menjawab ya atau tidak

pada setiap pertanyaan kemudian menekan tombol lanjut. Jika pengguna memilih pertanyaan, akan diarahkan ke gejala lainnya yang berhubungan dengan suatu penyakit yang sesuai sampai menuju pada kesimpulan hasil deteksi dan beralih ke halaman *Hasil Deteksi*. Gambar 5 dan Gambar 6 menunjukkan kasus pengguna memilih gejala yang dirasakannya dan kesimpulan hasil deteksi yang diberikan oleh aplikasi.

Kasus yang ditunjukkan dalam Gambar 5 dan 6 adalah pengguna aplikasi memilih beberapa pilihan gejala yang dirasakannya yaitu demam, keringat dingin waktu malam hari, lemah letih lesu tidak nafsu makan, batuk berdahak yang berlangsung lebih dari 14 hari, batuk yang mengeluarkan darah, terasa sesak dan nyeri pada dada. Berdasarkan aturan produksi pada Tabel 2 diperoleh penyakit yang memiliki gejala-gejala tersebut yaitu tuberkulosis paru (Gambar 6). Jumlah gejala yang dipilih yang sama dengan gejala tuberkulosis paru adalah 6 dari total 7 gejala dalam penyakit tersebut. Persentase yang ditunjukkan oleh aplikasi terhadap masukan pengguna adalah 85,71% yang diperoleh dari perhitungan probabilitas klasik sesuai Persamaan 1, yaitu $6/7 \times 100\%$.

Pengujian sistem menggunakan pengujian fungsional secara *black box* dan langsung oleh pakar. Pada pengujian oleh pakar dilakukan skenario kasus saat pengguna memilih gejala tertentu yang akan menghasilkan kesimpulan dari penyakit yang diderita. Rencana pengujian fungsional aplikasi sistem pakar deteksi dini tuberkulosis ini dapat dilihat pada Tabel 3. Hasil pengujian fungsional

halaman utama pada aplikasi sistem pakar ini ditunjukkan oleh Tabel 4. Hasil pengujian halaman deteksi yang dilakukan oleh sistem ini ditunjukkan oleh Tabel 5. Hasil pengujian fungsi halaman hasil deteksi pada aplikasi sistem pakar ditunjukkan oleh Tabel 6. Hasil pengujian fungsi halaman informasi penyakit ditunjukkan dalam Tabel 7. Keseluruhan fungsi sistem pakar telah dapat berjalan dengan baik secara fungsional.

Pengujian pakar pada aplikasi sistem pakar dilakukan untuk memvalidasi hasil diagnosis. Pengujian pakar dilakukan secara objektif di mana aplikasi diuji secara langsung. Pengujian pakar dilakukan oleh dr. Fathur Nur Kholis, Sp.PD. dari bagian S.M.F. Penyakit Dalam, RSUP Dr. Kariadi Semarang. Hasil pengujian oleh pakar ditunjukkan pada Gambar 7. Hasil pengujian ini dibandingkan dengan hasil pada Gambar 5 dan Gambar 6 untuk memberikan diagnosis terhadap penyakit tuberkulosis paru dengan gejala demam, keringat dingin waktu malam hari, lemah letih lesu tidak nafsu makan, batuk berdahak yang berlangsung lebih dari 14 hari, batuk yang mengeluarkan darah, terasa sesak dan nyeri pada dada. Pakar membenarkan diagnosis sistem pakar sesuai dengan gejala-gejala yang timbul dalam kasus pengujian.

Sistem pakar telah ini dapat berfungsi untuk mendeteksi dini tuberkulosis seperti halnya [7]-[11], khususnya tuberkulosis paru, tuberkulosis tulang, dan tuberkulosis kelenjar. Namun, sistem ini telah berjalan di perangkat Android, seperti halnya [4]-[5], sehingga dapat dijalankan secara *mobile* dan memudahkan pengguna untuk menggunakannya. Metode *forward chaining* yang digunakan dalam aplikasi telah divalidasi oleh seorang pakar [9], [11]. Tidak seperti penggunaan probabilitas klasik untuk menghitung persentase gejala yang dipilih untuk suatu penyakit, penggunaan CF seperti dalam [7], [10] atau metode Dempster Shafer, seperti dalam [12], dapat diterapkan untuk mendeteksi tuberkulosis paru, tulang, dan kelenjar sehingga memberikan jangkauan keyakinan pengguna dalam memberikan informasi gejala yang dirasakannya.

4. Kesimpulan

Sistem pakar pendeteksi penyakit tuberkulosis berbasis Android dengan menggunakan metode *forward chaining* telah berhasil dikembangkan. Sistem pakar ini telah dapat memberikan deteksi dini penyakit tuberkulosis paru, tuberkulosis tulang, dan tuberkulosis kelenjar berdasarkan 18 gejala yang menyertai ketiga penyakit tersebut. Persentase gejala yang dirasakan pengguna terhadap total gejala yang mungkin untuk suatu penyakit telah dapat disajikan dengan probabilitas klasik. Validasi oleh satu orang pakar telah memberikan hasil sesuai dengan yang diharapkan. Validasi oleh lebih dari satu pakar masih harus dilakukan untuk memberikan tingkat akurasi deteksi yang lebih baik.

5. Persantunan

Terima kasih diucapkan kepada dr. Fathur Nur Kholis, Sp. PD. yang telah membantu dalam menyelesaikan penelitian ini dan berperan sebagai pakar.

6. Daftar Pustaka

- [1] A.G. Icksan, and R. Luhur,S. Iwan, "Radiologi Toraks Tuberkulosis Paru ," CV. Agung Seto, Jakarta, 2008.
- [2] D. Sarwani, S. Nurlaela, and Z. A. Isnani, "Risk Factors of Multidrug Resistant Tuberculosis (MDR-TB)," Jurnal Kesehatan Masyarakat, vol. 8, no. 1, pp. 60–66, 2012.
- [3] R. A. Werdhani, "Patofisiologi, Diagnosis, dan Klafikasi Tuberkulosis," Departemen Ilmu Kedokteran Komunitas, Okupasi, dan Keluarga FKUI, 2002.
- [4] M.P. Hardiyanti, R. R. Isnanto, and I. P. Windasari, "Aplikasi Sistem Pakar Berbasis Mobile untuk Diagnosis Dini Meningitis," Jurnal Teknologi dan Sistem Komputer, [Online] vol. 5(2), pp. 83-88, doi:10.14710/jtsiskom.5.2.2017.84-89, 2017
- [5] M. P. N. Saputri, R. R. Isnanto, and I. P. Windasari,. Aplikasi Sistem Pakar Pendeteksi Gastroenteritis Berbasis Android. Jurnal Teknologi dan Sistem Komputer, [Online] vol. 5(3), pp. 110-114, doi:10.14710/jtsiskom.5.3.2017.110-11, 2017.
- [6] B. A. Herlambang, and V. A. V. Setyawati, "Perancangan Data Flow Diagram Sistem Pakar Penentuan Kebutuhan Gizi Bagi Individu Normal Berbasis Web," Jurnal Informatika Upgris, vol. 1(1), 2015.
- [7] Y. P. Bria, and E. A. S. Takung, "Pengembangan Sistem Pakar Diagnosis Penyakit Tuberculosis dan Demam Berdarah Berbasis WEB Menggunakan Metode Certainty Factor," Prosiding Seminar Nasional Teknologi Informasi dan Komunikasi, pp. 271–276, 2015.
- [8] N. Aini, R. Ramadiani, and H.R. Hatta, "Sistem Pakar Pendiagnosa Penyakit Tuberculosis," Informatika Mulawarman, vol. 12(1), pp.56-63, 2017.
- [9] M. Arsyad, "Implementasi Metode Sugeno Pada Sistem Pakar Penentuan Stadium Pada Penyakit Tuberculosis (TBC)," Pelita Informatika Budi Darma, vol. 7(3), pp. 22–27, 2014.
- [10] W. Widiastuti, D.D.S. Fatimah, and D.J. Damiri, "Aplikasi Sistem Pakar Deteksi Dini Pada Penyakit Tuberculosis," Jurnal Algoritma, vol. 9(01), 2012.
- [11] E.D.S. Mulyani, and I.N. Restianie, "Aplikasi Sistem Pakar untuk Mendiagnosa Penyakit Anak (Balita) dengan Menggunakan Metode Forward Chaining," SEMNASTEKNOMEDIA ONLINE, 4(1), pp. 43-47, 2016.
- [12] E. Lestari, and E. U. Artha, "Sistem Pakar dengan Metode Dempster Shafer untuk Diagnosis Gangguan Layanan Indihome di PT Telkom Magelang," Khazanah Informatika, vol. 3(1), pp. 16-24, 2017.

Mitra Bestari (Reviewer)

Dewan editor mengucapkan terima kasih yang sebesar-besarnya atas partisipasi para reviewer berikut ini yang telah membantu dalam proses penerbitan jurnal Khazanah Informatika volume 4 tahun 2018.

1. Jan Wantoro, UMS
2. Dr. Endah Sudarmilah, UMS
3. Tati Ernawati, Politeknik TEDC Bandung
4. Adi Supriyatna, AMIK BSI Karawang
5. Sitaresmi Wahyu Handani, AMIKOM Purwokerto
6. Dwi Ely Kurniawan, Politeknik Negeri Batam
7. Yusuf Sulisty Nugroho, UMS
8. Dedi Gunawan, UMS
9. Endang Wahyu Pamungkas, UMS
10. Umi Fadlilah, UMS
11. Yogiek Indra Kurniawan, UMS
12. Dr. Bana Handaga, UMS
13. Dr. Heru Supriyono, UMS
14. Sukirman, UMS