

K

Volume 3 No 2
Oktober 2014

Jurnal Sistem Komputer UNIKOM

Komputika

SRI SUPATMI, TAUFIK NUZWIR NIZAR, REZA FAHLEVI

Sistem Kontrol Peralatan Rumah dan Monitoring Kondisi Rumah melalui Internet Berbasis Web dan Open WRT

FOLKES E. LAUMAL

Implementasi Pengembangan Penjadwalan Round Robbin pada Antrian Data Real Time Central Processing Unit

ZAENAL ABIDIN, SUSMINI INDRIANI LESTARININGATI

Sistem Keamanan dan Monitoring Rumah Pintar secara Online Menggunakan Perangkat Mobile

SRI NURHAYATI, RYAN ZULMI

Sistem Monitoring Distribusi Bantuan Bencana Alam Berbasis Web menggunakan Metode Algoritma First-Fit

FEBRYANTI STHEVANIE

Deteksi Lokal Tampering pada Video Menggunakan Algoritma LBP-TOP

SELVIA LORENA BR. GINTING, WENDI ZARMAN, ASTRID DARMAWAN

Teknik Data Mining untuk Memprediksi Masa Studi Mahasiswa Menggunakan Algoritma K-Nearest Neighborhood

Diterbitkan Oleh:
Jurusan Teknik Komputer
UNIKOM



KOMPUTIKA (JURNAL SISTEM KOMPUTER UNIKOM)

Volume 3 No. 2, Oktober 2014

ISSN 2252-9039

Pelindung

Dr. Ir. Eddy Soeryanto Soegoto

Ketua Penyunting

Susmini I. Lestaringati, S.T, M.T.

Editor Pelaksana

Agus Mulyana, M.T.

Usep Mohamad Ishaq, M.Si.

Aprianti Putri Sujana, S.Kom, M.T

Editor Tamu

Dr. Arry Akhmad Arman, ITB

Ir. Aldo Agusdian, M.T, ITB

Mitra Bestari

Dr. Wendi Zarman, M.Si.

Dr. Yeffry Handoko Putra, M.T

Ir. Syahrul, M.T.

Pelaksana Teknis

Ayub Subandi, M.T.

Jurnal ini terbit dua kali setahun

Setiap Bulan April dan Oktober

Alamat Redaksi

Jurusan Teknik Komputer

Universitas Komputer Indonesia

Jl. Dipati Ukur 112

Bandung 40132

Tel/Fax : (022) 2504119/ (022) 2533754

<http://komputika.tk.unikom.ac.id>

DAFTAR ISI

Sistem Kontrol Peralatan Rumah dan Monitoring Kondisi Rumah Melalui Internet Berbasis Web dan OpenWRT Sri Supatmi, Taufiq Nuzwir Nizar, Reza Fahlevi	1 - 6
Implementasi Pengembangan Penjadwalan Round Robbin Pada Antrian Data Real Time Central Processing Unit Folkes E. Laumal	7 - 12
Sistem Keamanan dan Monitoring Rumah Pintar Secara Online Menggunakan Perangkat Mobile Zainal Abidin, Susmini Indriani Lestaringati	13 - 17
Sistem Monitoring Distribusi Bantuan Bencana Alam Berbasis Web Menggunakan Metode Algoritma First-Fit Sri Nurhayati, Ryan Zulmi	18 - 22
Deteksi Local Tampering Pada Video Menggunakan Algoritma LBP-TOP Febryanti Sthevanie	23 - 28
Teknik Data Mining Untuk Memprediksi Masa Studi Mahasiswa Menggunakan Algoritma K-Nearest Neighborhood Selvia Lorena Br. Ginting, Wendi Zarman, Astrid Darmawan	29 - 34

SISTEM KONTROL PERALATAN RUMAH DAN MONITORING KONDISI RUMAH MELALUI INTERNET BERBASIS WEB DAN OPENWRT

Sri Supatmi¹, Taufiq Nuzwir Nizar², Reza Fahlevi³

^{1,2,3}Jurusan Teknik Komputer Universitas Komputer Indonesia (UNIKOM), Bandung

¹sri.supatmi28@email.unikom.ac.id, ²taufiq.nizar@gmail.com, ³rezafahlevi08@gmail.com

ABSTRAK

Pengontrolan dan monitoring peralatan rumah tangga berupa tampilan suhu ruangan, dan visualisasi kondisi ruangan secara real time akan sangat membantu pemilik rumah jika dilakukan melalui media internet. Hal ini memungkinkan bagi pemilik rumah untuk mengetahui kondisi rumahnya dari jarak yang jauh. Kondisi rumah dapat diketahui jika terdapat koneksi langsung dari pemilik rumah melalui internet dengan sistem yang berada di rumah. Koneksi internet membutuhkan IP Public, Personal Computer (PC) server dan IP kamera yang harganya masih relatif mahal untuk suatu sistem yang diaplikasikan pada sebuah rumah. Permasalahan diatas mendasari dilakukannya penelitian berupa perancangan sistem kontrol peralatan rumah yang sekaligus memonitor kondisi rumah. Sistem ini memanfaatkan web hosting, koneksi internet yang fleksibel melalui media kabel maupun media radio, dan juga aplikasi berbasis web untuk fleksibilitas pengaksesan sistem yang digunakan sebagai perantara bagi sistem yang dirancang dengan pemilik rumah. Pada perancangan ini memanfaatkan TL-MR3020 sebagai pengganti PC Server di rumah. Pengujian dilakukan untuk menghasilkan waktu proses pada pengontrolan peralatan rumah berupa lampu dan data. Data dari sistem yang dibuat berupa data sensor suhu, status lampu ruangan dan tampilan visual ruangan rumah. Hasil pengujian pengontrolan peralatan rumah didapatkan waktu rata-rata 11.5 detik. Sedangkan hasil pengujian berupa pembacaan data suhu, status lampu ruangan dan tampilan visual ruangan rumah didapatkan waktu rata-rata 11.6 detik.

Kata kunci : smart home system, OpenWrt, Web Hosting, TL-MR3020.

1. PENDAHULUAN

Memonitor kondisi rumah atau mengontrol peralatan rumah tangga seperti lampu masih dilakukan secara manual, sehingga ketika pemilik rumah sedang bepergian atau rumah dalam keadaan kosong, pemilik rumah tidak dapat mengetahui kondisi dan mengontrol peralatan rumah.

Dari permasalahan tersebut, penelitian yang dilakukan bertujuan membuat sistem kontrol peralatan rumah dan monitoring kondisi rumah secara jarak jauh berbasis *web* dan *openWrt*. Sistem berbasis web dan *openWrt* ini merupakan salah satu solusi yang memudahkan bagi pemilik rumah mengontrol peralatan rumah berupa lampu dan memonitor kondisi rumahnya secara jarak jauh melalui internet dengan perangkat yang memiliki akses internet tanpa membutuhkan dana yang besar untuk mengaplikasikan sistem tersebut.

2. TUJUAN

Penelitian ini bertujuan untuk melakukan monitoring dan mengontrol peralatan rumah tangga dengan memanfaatkan web hosting dan TL-MLR3020.

3. DASAR TEORI

3.1. Internet

Internet adalah kumpulan berbagai macam sistem jaringan komputer di dunia yang terkoneksi satu sama lain dan dapat saling berkomunikasi satu sama lain.

3.2. Web Hosting

Layanan *web hosting* adalah salah satu jenis layanan internet hosting yang memungkinkan individu dan organisasi untuk membuat website mereka dapat diakses melalui *World Wide Web*. *Web host* adalah perusahaan yang menyediakan ruang pada *server* yang dimiliki atau disewa untuk digunakan oleh klien, serta menyediakan konektivitas internet, biasanya di sebuah pusat data. Ruang lingkup layanan *web hosting* sangat bervariasi. Yang paling mendasar adalah halaman *web* dan *hosting file* skala kecil, di mana *file* dapat di-unggah melalui *File Transfer Protocol* (FTP) atau antarmuka *Web*.

3.3. OpenWrt

OpenWrt adalah sistem operasi / sistem operasi tertanam berbasis kernel Linux, dan utamanya digunakan pada perangkat tertanam untuk merutekan lalu lintas jaringan. Komponen

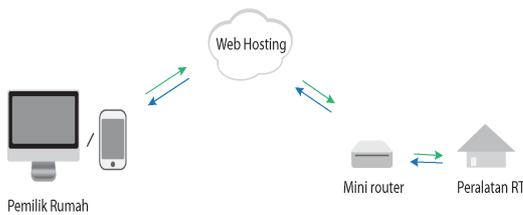
utamanya adalah kernel Linux, util-linux, uClibc dan BusyBox. Semua komponen telah dioptimasi dalam hal ukuran, agar cukup kecil untuk pas pada ketersediaan media penyimpanan dan memori yang terbatas dalam *router* rumah.

4. PERANCANGAN SISTEM

Perancangan sistem dilakukan dalam beberapa tahapan diantaranya perancangan aplikasi untuk pemilik rumah, perancangan aplikasi pada *mini router* yang berada di rumah dan perancangan aplikasi pada *web hosting* yang akan menjadi perantara antara pemilik rumah dan rumah yang akan dikontrol dan dipantau.

4.1. Diagram Blok

Gambar 1 menjelaskan tentang diagram blok sistem secara umum. Diagram blok terdiri dari dua sisi yang akan saling berkomunikasi untuk tujuan kontrol dan monitoring dengan web hosting sebagai perantaranya.



Gambar 1 Diagram Blok Sistem

Pemilik rumah yang berada di tempat yang jaraknya jauh dari rumah dapat mengontrol peralatan rumah tangganya atau memonitor keadaan rumah dengan menggunakan komputer maupun perangkat *mobile* yang memiliki akses internet.

4.2. Aplikasi Perangkat Pemilik Rumah

Pemilik rumah memerlukan suatu perangkat yang dapat digunakan untuk mengakses sistem yang dirancang yaitu berupa aplikasi *web* yang berada di *server hosting*. Pemilik rumah dapat menggunakan komputer, laptop, maupun perangkat *mobile* miliknya. Aplikasi yang harus terdapat di dalam perangkat tersebut adalah peramban web versi terbaru yang mendukung teknologi *Server-Sent Event*.

Server-sent DOM events - Candidate Recommendation

Method of continuously sending data from a server to the browser, rather than repeatedly requesting it (EventSource interface, used to fall under HTML5)

Global user stats: Support: 65.14% Partial support: 0.05% Total: 65.19%

Resources: [Browsers with demo](#) [HTML5 Rocks tutorial](#) [browsers](#)

	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	BlackBerry Browser	Opera Mobile	Chrome for Android	Firefox for Android	IE Mobile	
30 versions back													
29 versions back		2.0	5.0										
28 versions back		3.0	6.0										
27 versions back		3.5	7.0										
26 versions back		3.6	8.0										
25 versions back		4.0	9.0										
24 versions back		5.0	10.0										
23 versions back		6.0	11.0										
22 versions back		7.0	12.0										
21 versions back		8.0	13.0										
20 versions back		9.0	14.0										
19 versions back		10.0	15.0										
18 versions back		11.0	16.0										
17 versions back		12.0	17.0										
16 versions back		13.0	18.0		9.0								
15 versions back		14.0	19.0		9.5-9.6								
14 versions back		15.0	20.0		10.0-10.1								
13 versions back		16.0	21.0		10.5								
12 versions back		17.0	22.0		10.6								
11 versions back		18.0	23.0		11.0								
10 versions back		19.0	24.0		11.1								
9 versions back		20.0	25.0		11.5								
8 versions back		21.0	26.0		11.6								
7 versions back		22.0	27.0	3.1	12.0			2.1					
6 versions back	5.5	23.0	28.0	3.2	12.1			2.2			10.0		
5 versions back	6.0	24.0	29.0	4.0	15.0	3.2		2.3			11.0		
4 versions back	7.0	25.0	30.0	5.0	16.0	4.0-4.1		3.0			11.1		
3 versions back	8.0	26.0	31.0	5.1	17.0	4.2-4.3		4.0			11.5		
2 versions back	9.0	27.0	32.0	6.0	18.0	5.0-5.1		4.1			12.0		
Previous version	10.0	28.0	33.0	6.1	19.0	6.0-6.1		4.2-4.3	7.0		12.1		
Current	11.0	29.0	34.0	7.0	20.0	7.0	5.0-7.0	4.4	10.0	16.0	33.0	26.0	10.0
Near future		30.0	35.0		21.0								
Farther future		31.0	36.0		22.0								

Gambar 2 Peramban Yang Mendukung SSE

4.3. Perancangan Aplikasi Web

Pada sistem ini *web hosting* berfungsi sebagai perantara antara pemilik rumah dan peralatan yang berada di rumah.

Antarmuka antara pemilik rumah dan sistem dirancang sebuah halaman *web* untuk mengontrol peralatan rumah tangga dan juga memantau keadaan rumahnya dengan gambar yang selalu diperbaharui.

Pemilik rumah dapat membuka *web* dari komputer maupun perangkat *mobile*-nya maka antarmuka sistem didesain dengan *responsive web design* (RWD) untuk menyesuaikan tampilan sistem pada layar komputer maupun layar perangkat *mobile*.

Ketika *website* dibuka di peramban web *mobile* yang ukuran layarnya lebih kecil maka aplikasi mampu menyesuaikan dengan ukuran layar perangkat tersebut. Gambar 4 adalah contoh tampilan halaman login yang menyesuaikan dengan layar perangkat *mobile*.

4.4. Proses Login dan Logout

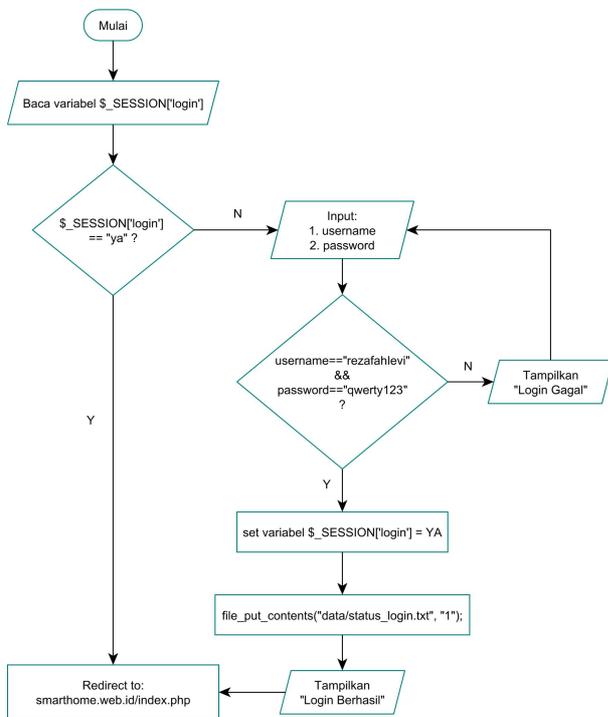
Proses Login dilakukan dengan cara memasukkan *username* dan *password* dengan benar oleh pemilik rumah. Jika *username* dan *password* yang dimasukkan benar, maka sistem akan masuk ke halaman utama.

Jika pemilik rumah belum melakukan *login* atau melakukan *logout* sebelumnya maka akan ditampilkan form untuk mengisi *username* dan *password* yang akan diperiksa oleh program, jika salah satu masukan salah maka tampilkan keterangan Login Gagal.

Jika *username* dan *password* benar maka set variabel `$_SESSION` dengan karakter "YA".

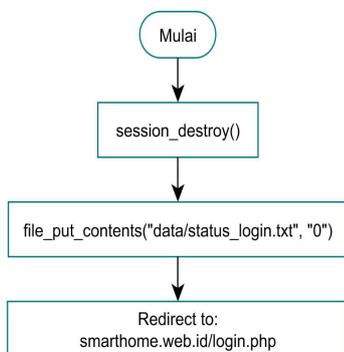
Sistem Kontrol Peralatan Rumah Dan Monitoring Kondisi Rumah Melalui Internet Berbasis Web dan OpenWrt

Setelah itu isi dokumen *txt* dengan karakter “1”, dokumen *txt* ini akan diakses oleh mini server untuk mengetahui apakah pemilik rumah sedang *login* atau tidak dan menyesuaikan proses yang dilakukan. Setelah proses-proses tersebut selesai maka tampilkan keterangan **Login Berhasil** dan arahkan ke halaman utama.



Gambar 3 Proses Login

Proses logout dijalankan jika pemilik rumah menekan tombol logout pada halaman utama. Proses dimulai dengan menghapus isi variabel `$_SESSION` yang sebelumnya berisi karakter “1”. Kemudian ganti isi dokumen *txt* dengan karakter “0” yang berarti pemilik rumah tidak sedang login.



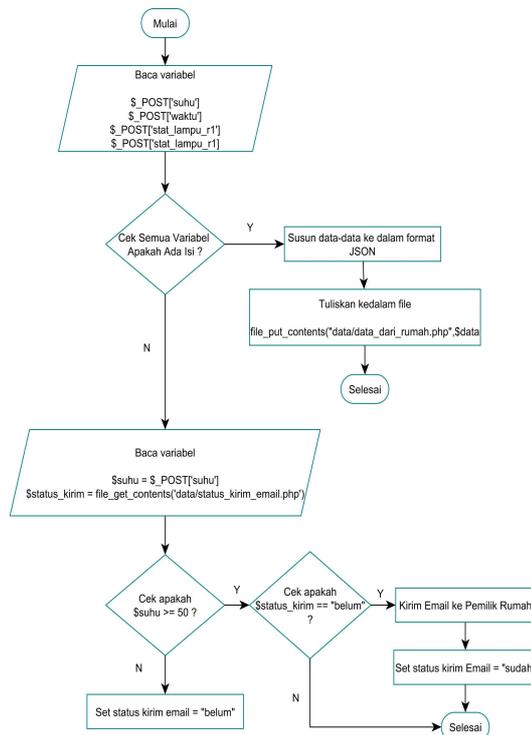
Gambar 4 Proses Logout

4.5. Proses Data kirim dari Rumah

Diagram alir berikut berada di web hosting dan akan bekerja jika mendapat POST dari server rumah yang mengirimkan data-data.

Pada diagram alur gambar 5 berikut dijelaskan pada awalnya program akan memeriksa variabel suhu, waktu, status lampu ruang 1, dan status lampu ruang 2 jika semua variabel berisi data maka susun data-data tersebut ke dalam format JSON lalu tuliskan kedalam file yang disimpan di

`http://myhome.web.id/data/data_dari_rumah.php` p dengan fungsi php `file_put_content`.



Gambar 5 Proses Data Rumah

Jika pemilik rumah tidak login maka aplikasi hanya menampilkan data suhu dan mengirimkan data suhu yang terbaca. Suhu yang terbaca dibandingkan dengan suhu pembanding dalam sistem. Suhu pembanding diset dengan nilai 50°C.

Jika data suhu yang dikirim lebih besar atau sama dengan 50°C kirimkan email peringatan ke alamat email pemilik rumah. Sebelum mengirim email akan dilakukan pemeriksaan apakah sebelumnya email sudah dikirimkan, hal ini untuk mencegah pengiriman email berkali-kali.

4.6. Proses Data Perintah Kontrol Lampu

Pemilik rumah dapat melakukan kontrol kepada peralatan rumah tangganya dalam hal ini

lampu yang berada pada ruangan yang berbeda dengan menekan tombol perintah yang disediakan.



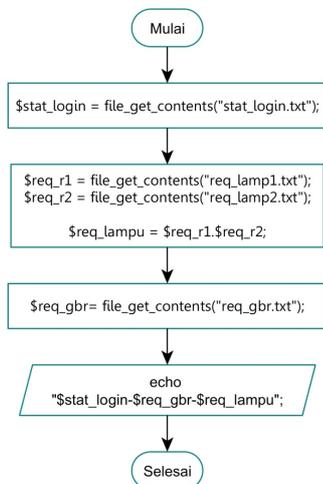
Gambar 6 Tombol Perintah Kontrol Lampu

Pada gambar 7 menunjukkan tombol perintah yang dapat dipilih pemilik rumah untuk melakukan kontrol terhadap lampu yang diinginkan. Pada ruang 1 terdapat tombol **ON** untuk mengirim perintah menyalakan lampu, **OFF** untuk mengirim perintah mematikan lampu, sedangkan **AUTO** digunakan untuk mengirim perintah mematikan atau menyalakan lampu berdasarkan sensor cahaya.

4.7. Penggabungan Data Request

Gambar 7 menjelaskan bagaimana cara penggabungan data yang dibutuhkan untuk diakses secara berkala oleh mini router yang berada di rumah. Adapun data yang diperlukan oleh mini router adalah:

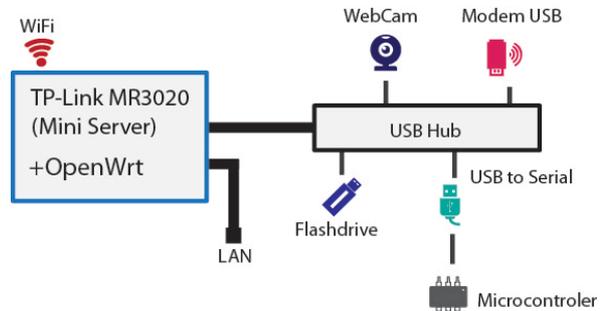
1. Status login pemilik rumah (lokasi: /data/stat_login.txt)
2. Data permintaan untuk update gambar terbaru (lokasi: /data/req_gbr.txt)
3. Data permintaan untuk kontrol lampu (lokasi: /data/req_lamp1.txt dan /data/req_lamp2.txt)



Gambar 7 Penggabungan Data

4.8. Perancangan Sistem Pada Server Rumah

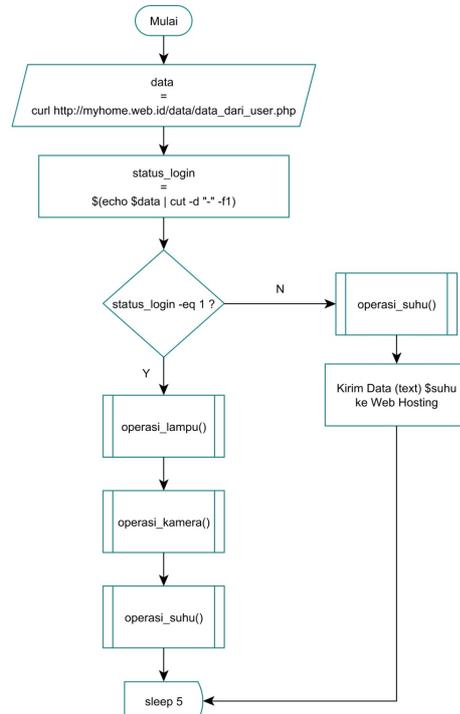
Sistem yang akan dirancang dalam mini router akan bertugas untuk mengambil perintah yang berada di *web hosting* dan memproses perintah tersebut lalu meneruskannya ke perangkat-perangkat yang terhubung ke mini server dapat dilihat pada Gambar 8.



Gambar 8 Server Rumah dan Perangkat Pendukung

4.9. Aplikasi Utama Pada Server Rumah

Di dalam mini server akan dirancang sebuah aplikasi memanfaatkan pemrograman bash yang akan melakukan proses pemeriksaan data permintaan pemilik rumah yang tersimpan pada *web hosting* lalu memproses data tersebut.



Gambar 9 . Diagram Alir Aplikasi Utama

4.10. Operasi Suhu

Gambar 10 adalah fungsi operasi_suhu() akan dijalankan saat dipanggil pada program utama.

Sistem Kontrol Peralatan Rumah Dan Monitoring Kondisi Rumah Melalui Internet Berbasis Web dan OpenWrt

Pada fungsi ini akan memanfaatkan program aplikasi python yang sudah tersedia untuk dipanggil atau digunakan sebagai komunikasi ke perangkat keras.



Gambar 10 Operasi Suhu

4.11. Operasi Lampu

Operasi dimulai dengan membaca data perintah untuk kontrol lampu dengan *tool CUT* pada *field 3(-f3)* seperti yang dijelaskan pada format data di atas lalu menyalin data tersebut kedalam variabel *\$req_lampu*. Setelah variabel tersebut terisi dengan karakter berupa perintah maka lakukan pemeriksaan isi variabel tersebut seperti pada tabel berikut.

Tabel 1 Tampilan Status Lampu

var <i>Sreq_lampu</i>	Aksi Untuk Lampu	
	ruang 1	ruang 2
lon2on	ON	ON
lon2off	ON	OFF
lon2dim	ON	REDUP
loff2on	OFF	ON
loff2off	OFF	OFF
loff2dim	OFF	REDUP
lauto2on	AUTO	ON
lauto2off	AUTO	OFF
lauto2dim	AUTO	REDUP

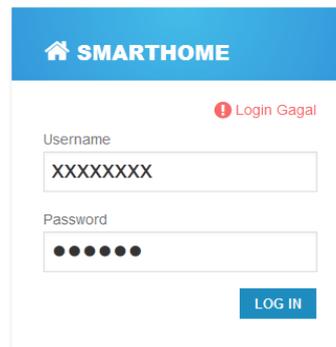
5. HASIL PENGUJIAN

Pengujian dilakukan untuk mengetahui hasil perancangan yang dibuat apakah sesuai dengan tujuan awal yang ingin dicapai. Pengujian meliputi pengujian kontrol terhadap lampu rumah, penerimaan data-data dari rumah, dan

pengiriman email peringatan kepada pemilik rumah.

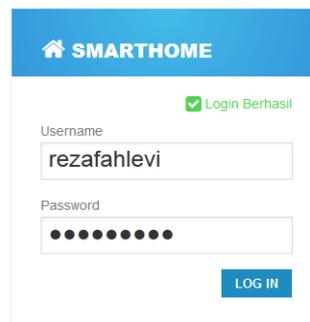
5.1. Pengujian Login di Sistem

Pengujian dilakukan pada sistem login pada program php yang telah dirancang sebelumnya dan tersimpan di web hosting. Proses login berfungsi untuk mencegah orang yang tidak berhak memanfaatkan sistem yang telah dibuat dan dapat masuk pada halaman utama. Pengujian pertama dilakukan dengan cara memasukkan username dan password yang bukan merupakan username dan password yang diizinkan untuk login. Tampilan Login dapat dilihat pada Gambar 11 dan Gambar 12



Gambar 11 Hasil Pengujian Login Dengan Username Yang Salah

Pada hasil pengujian di atas sistem berhasil untuk tidak mengizinkan sembarang username untuk masuk ke sistem utama dan menampilkan keterangan Login Gagal.



Gambar 12 Hasil Pengujian Login Dengan Username Yang Benar

Pada hasil pengujian kedua di atas dimasukkan *username* dan *password* yang sesuai dengan *username* dan *password* yang telah ditentukan di dalam program untuk bisa masuk ke halaman utama. Saat tombol login ditekan maka akan ditampilkan keterangan Login Berhasil dan akan di-*redirect* ke halaman utama.

5.2. Pengujian Kontrol Terhadap Lampu

Pada pengujian ini akan dilakukan kontrol terhadap lampu yang berada pada ruang 1 dan ruang 2 dengan cara menekan tombol perintah pada halaman web. Untuk mengetahui waktu rata-rata pemrosesan perintah kontrol, waktu pemrosesan akan dihitung mulai dari tombol perintah ditekan sampai dengan hasil yang didapatkan. Setiap tombol perintah akan diuji satu persatu apakah hasilnya berhasil atau tidak.

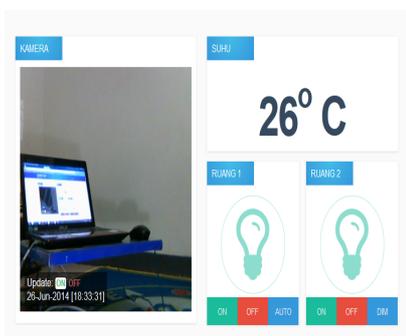
Tabel 2 Hasil Pengujian Kontrol Lampu

No	Perintah	Hasil	Waktu(s)
1	R1 ON	Lampu R1 Menyala	20
2	R1 OFF	Lampu R1 Padam	18
3	R1 AUTO	Lampu R1 Otomatis	9
4	R2 ON	Lampu R2 Menyala	5
5	R2 OFF	Lampu R2 Padam	6
6	R2 DIM	Lampu R2 Redup	11
Rata-rata			11,5

Tabel 2 adalah hasil pengujian untuk pengontrolan lampu rumah. Perbedaan waktu yang digunakan untuk pengontrolan lampu dipengaruhi oleh beberapa faktor yaitu kecepatan internet untuk melakukan akses di web hosting. Waktu rata-rata yang dibutuhkan untuk pengontrolan lampu adalah 11,5 detik.

5.3. Pengujian Penerimaan Data Dari Rumah

Gambar 13 menjelaskan tentang waktu yang diperlukan untuk perubahan data kondisi rumah.



Gambar 13 Tampilan Penerimaan Data

Tabel 3 adalah hasil pengujian berupa perubahan waktu untuk pembacaan kondisi rumah.

Tabel 3 Waktu untuk pengiriman data

No	Hasil	Waktu(s)
1	Data Terbaru Ditampilkan	9
2	Data Terbaru Ditampilkan	10
3	Data Terbaru Ditampilkan	16
4	Data Terbaru Ditampilkan	10
5	Data Terbaru Ditampilkan	10
6	Data Terbaru Ditampilkan	16
7	Data Terbaru Ditampilkan	10
8	Data Terbaru Ditampilkan	10
9	Data Terbaru Ditampilkan	15
10	Data Terbaru Ditampilkan	10
Waktu Rata-rata		11.6

6. KESIMPULAN

Setelah dilakukan pengujian sistem, maka didapatkan beberapa kesimpulan sebagai berikut:

1. Web hosting dan TL-MR3020 telah dapat berfungsi dengan baik sebagai perantara antara pemilik rumah dan server rumah.
2. Pengontrolan lampu rumah dan pemantauan kondisi rumah berhasil dilakukan dengan waktu proses rata-rata 11.5 detik.

7. DAFTAR PUSTAKA

- [1] *Antarmuka Pemrograman Aplikasi*. Diakses pada tanggal 15 April 2014, dari http://id.wikipedia.org/wiki/Antarmuka_pemrograman_aplikasi..
- [2] *Cascading Style Sheets*. Diakses pada tanggal 15 April 2014, dari http://id.wikipedia.org/wiki/Cascading_Style_Sheets.
- [3] Clark, M. P. (2003). *Data Networks, IP and the Internet - Protocols, Design and Operation*. Chichester: John Wiley & Sons Ltd.
- [4] Grigorik, I. (2013). *High-Performance Browser Networking*. Sebastopol: O'Reilly Media, Inc.
- [5] *OpenWrt Version History*. Diakses pada tanggal 20 April 2014, dari <http://wiki.openwrt.org/about/history>.
- [6] Sidik, B. (2012). *Pemrograman Web Dengan PHP*. Bandung: Informatika Bandung.
- [7] Sofana, I. (2012). *CISCO CCNA & Jaringan Komputer*. Bandung: Informatika Bandung.
- [8] Suryana, T., & Sarwono, J. (2007). *Membuat Web Probadi dan Bisnis dengan HTML*. Yogyakarta: Gava Media

IMPLEMENTASI PENGEMBANGAN PENJADWALAN *ROUND-ROBBIN* PADA ANTRIAN DATA *REAL TIME CENTRAL PROCESSING UNIT*

Folkes E. Laumal

Politeknik Negeri Kupang, Jl. Adicucipto – Penfui - Kupang, Kode Pos 85361

Email : folkeslaumal76@gmail.com

ABSTRAK

Penjadwalan merupakan dasar sistem operasi komputer multiprogramming yang membuat sistem menjadi lebih produktif dan efisien. Tujuan dari penjadwalan proses adalah meningkatkan kinerja sistem komputer menurut kriteria tertentu diantaranya *waiting time* dan *turn around time*. *Real time fetching* adalah proses pengambilan data dari antrian pada memory secara *real time*[1]. Salah satu penjadwalan yang dapat diterapkan dalam pengolahan sistem operasi yaitu *Round Robbin*. Penjadwalan ini bersifat *preemptive* sehingga ketika diterapkan pada *real time fetching* diperoleh hasil *performance* yang rendah[5]. Untuk itu telah dikembangkan lagi sebuah algoritma *Round Robbin* (*Pengembangan Round Robbin*) dan diuji dalam 5 antrian program. Hasil akhir membuktikan bahwa dengan algoritma pengembangan *Round Robbin* tersebut telah memperkecil *waiting time (AWT)* dari 7.8 ke 3.6 satuan waktu dan 8.6 ke 6 satuan waktu. *Turn around time (ATAT)* berkurang dari 13.4 ke 8 satuan waktu dan 12.2 ke 11.6 satuan waktu. Sedangkan *performance CPU* telah meningkat dari 0.128 ke 0.278 dan dari 0.116 ke 0.167.

Kata kunci : *Round-Robbin, waiting time, turn around time, Performance CPU*

1. PENDAHULUAN

Penjadwalan merupakan dasar dari sistem operasi (OS) dalam *multiprogramming computer* karena dengan mengatur aliran dari proses-proses yang ada pada sebuah antrian sistem operasi akan membuat sistem komputer menjadi lebih produktif dan efisien. Sasaran *multiprogramming* adalah mempunyai proses yang tereksekusi di setiap waktu sehingga *utilisasi* dari pemroses menjadi lebih baik. Untuk sistem komputer dengan pemroses tunggal (*single processor*), tidak pernah lebih dari satu proses yang berjalan.

Di dalam sebuah sistem komputer, jika terdapat beberapa proses, maka satu proses akan berjalan dan proses lain akan menunggu sampai proses sebelumnya selesai dieksekusi. Ide *multiprogramming* sebenarnya sederhana karena satu proses dieksekusi sampai selesai tanpa menunggu, biasanya terjadi dalam operasi I/O.

Pada *multiprogramming*, dalam suatu waktu tertentu beberapa proses akan disimpan dimemori. Ketika proses tertentu harus menunggu dalam antrian, setiap kali satu proses harus menunggu, proses lain akan mengambil alih penggunaan pemroses. Dalam hal ini, sistem operasi mengambil pemroses darinya dan memberikan pemroses ke proses lain. Pola ini dilakukan secara terus-menerus.

Tujuan utama dari penjadwalan proses adalah meningkatkan kinerja sistem komputer

menurut kriteria tertentu. Kinerja untuk mengukur dan optimasi kerja penjadwalan adalah *fairness, response time, turn around time* dan *throughput*[3].

Terdapat 2 model penjadwalan dalam sistem operasi yaitu penjadwalan *preemptive* dan *non-preemptive*. Masing-masing penjadwalan itu memiliki sub model dengan spesifikasi dan implementasinya masing-masing, termasuk *Round Robin Scheduling* yang merupakan salah satu penjadwalan *preemptive* yang tanpa prioritas [5]. Penjadwalan ini banyak dikembangkan dalam sistem operasi modern saat ini ketika sebuah antrian dalam pemrosesan data CPU terjadi. Akan tetapi sebenarnya penjadwalan *Round robin* memiliki keterbatasan jika diterapkan pada sistem antrian yang *real time*. Pada kondisi ini antrian yang datang secara terus-menerus menyebabkan *time sharing* akan bertambah dan menyebabkan menurunnya *performance* dari CPU.

Untuk memperbaiki penurunan *performance* ini, Ajit Singh, Priyanka Goyal dan Sahil Bahtra (2012) telah berhasil menurunkan jumlah *switching* dan rata-rata waktu tunggu dengan pendekatan algoritma lain yang diuji pada kasus 5 buah antrian. Akan tetapi seberapa besar penurunan pada rata-rata waktu tunggu dan rata-rata *turn around time* sehingga dapat memberikan kesimpulan yang pasti tentang perbaikan *performance* ini, maka

penulis telah menguji kembali melalui sebuah algoritma pengembangan Round Robin dan menerapkan pada 5 antrian aritmetika pada CPU. Selanjutnya menghitung AWT dan ATAT dengan memperhitungkan *arrival time* dan tanpa *arrival time*. Hasil AWT dan ATAT ini kemudian dibandingkan dengan penjadwalan Round Robin umum dalam bentuk grafik.

2. Definisi Scedulling

Penjadwalan proses merupakan kumpulan kebijaksanaan dan mekanisme di sistem operasi yang berkaitan dengan urutan kerja yang dilakukan sistem komputer[5].

Penjadwalan bertugas memutuskan proses yang harus berjalan dan kapan dan selama berapa lama proses itu berjalan. Kriteria untuk mengukur dan optimasi kinerja sebuah penjadwalan adalah :*Fairnes* (adil), *Eeficiency* (efisiensi), *Response time* (waktu tanggap); yang terbagi dalam 2 sistem, yaitu Sistem interaktif dan Sistem waktu nyata, *Turn around time*; dengan persamaan waktu eksekusi + waktu menunggu serta *Throughput*[3]. Tujuan penjadwalan yaitu :

1. Menjamin tiap proses mendapat pelayanan dari pemroses yang adil.
2. Menjaga agar pemroses tetap dalam keadaan sibuk sehingga efisiensi mencapai maksimum. Pengertian sibuk adalah pemroses tidak menganggur, termasuk waktu yang dihabiskan untuk mengeksekusi program pemakai dan sistem operasi.
3. Meminimalkan waktu tanggap.
4. Meminimalkan turn around time.
5. Memaksimalkan jumlah job yang diproses persatu interval waktu. Lebih besar angka *throughput*, maka akan lebih banyak kerja yang dilakukan sistem [5].

3. Fetching data pada CPU

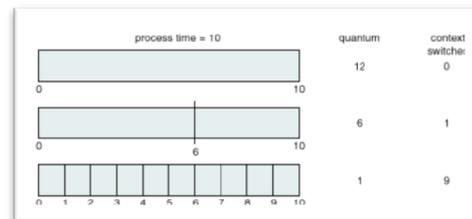
Fetching adalah proses pembacaan byte instruksi dari antrian memory menggunakan program counter (PC) sebagai alamat memory. Instruksi tersebut kemudian diekstrak menjadi 2 porsi 4-bit yang dikenal sebagai *icode* (*instruction code*) dan *ifun* (*instruction function*) [1]. Secara keseluruhan ada 5 langkah proses dalam organisasi processor, yaitu *fetching* (pembacaan data), *decode* (pengalamatan), *execute* (pengolahan/eksekusi), *memory* (penyimpanan) dan *write back* (kembali ke register file)[1]. Antrian yang tersusun dalam

processor ketika sebuah komputer On adalah antrian real time yang berasal dari semua operasi pengolahan data yang dikerjakan. Kelima proses ini berjalan secara paralel dalam mengeksekusi setiap pengolahan data dalam CPU. Apabila proses *fetching* lambat, maka proses selanjutnya juga ikut lambat dan dapat mempengaruhi *performance* sebuah CPU (*Central Processing Unit*). Padahal dalam menggunakan komputer, setiap user akan menginginkan *performance* komputer yang cepat dalam pekerjaannya.

4. Penjadwalan Round-Robin

Algoritma ini menggilir proses yang ada di antrian. Proses akan mendapat jatah sebesar *time quantum*. Jika *time quantum*-nya habis atau proses sudah selesai, CPU akan dialokasikan ke proses berikutnya. Tentu proses ini cukup adil karena tak ada proses yang diprioritaskan, semua proses mendapat jatah waktu yang sama dari CPU yaitu $(1/n)$, dan tak akan menunggu lebih lama dari $(n-1)q$ dengan q adalah lama 1 quantum.

Algoritma ini sepenuhnya bergantung besarnya *time quantum*. Jika terlalu besar, algoritma ini akan sama saja dengan algoritma *first come first served*. Jika terlalu kecil, akan semakin banyak peralihan proses sehingga banyak waktu terbuang.



Gambar 4.1. Penjadwalan Round-Robin

Permasalahan pada Round Robin adalah menentukan besarnya *time quantum*. Jika *time quantum* yang ditentukan terlalu kecil, maka sebagian besar proses tidak akan selesai dalam 1 quantum. Hal ini tidak baik karena akan terjadi banyak switch, padahal CPU memerlukan waktu untuk beralih dari suatu proses ke proses lain (*context switches time*). Sebaliknya, jika *time quantum* terlalu besar, algoritma Round Robin akan berjalan seperti algoritma *first come first served*. *Time quantum* yang ideal adalah jika 80% dari total proses memiliki CPU *burst time* yang lebih kecil dari 1 *time quantum*[4].

Implementasi Pengembangan Penjadwalan *Round-Robbin* pada Antrian Data *Real Time Central Processing Unit*

Sebelum dibuat pengembangan penjadwalan round robin dalam algoritma lain, penulis terlebih dahulu akan memaparkan algoritma umum yang biasa diberlakukan dalam penanganan proses CPU.

Algoritma penjadwalan round robin diberikan sebagai berikut [3]:

- a. Mengatur semua proses di dalam antrian
- b. Memberikan time quantum (q) untuk membatasi waktu proses
- c. Jika waktu proses selesai, proses ditunda dan ditambahkan pada antrian ready.
- d. Jika suatu proses memiliki burts time $>$ time quantum, maka proses akan melepaskan CPU setelah selesai dan CPU akan segera dapat digunakan oleh proses selanjutnya.
- e. Jika suatu proses memiliki burst time $<$ time quantum, maka proses tersebut akan dihentikan jika sudah mencapai waktu quantum dan selanjutnya mengantri kembali pada posisi ekor dari daftar antrian. CPU kemudian menjalankan proses berikutnya.
- f. Tidak ada proses yang menunggu lebih dari $(n-1)q$ unit waktu.
- g. Jika q besar maka round robin menggunakan aturan FCFS, sedangkan jika q kecil, maka menggunakan mekanisme konteks *switch*.

Untuk menerapkan Round Robbin pada proses *fetching*, disediakan sebuah program dengan 5 proses sebagai berikut :

```
P1 : pushl %edi
P2 : pushl %esi
P3 : imul 16(%ebp), %eax
P4 : leal 0(,%eax,4), %ecx
P5 : movl %ebx, %edx
```

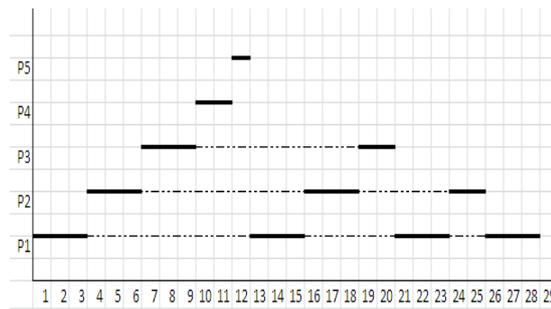
Jika diolah dengan algoritma Round Robbin umum sebagai berikut :

- a. Eksekusi dengan Burst Time (BT) saja.

Artinya ketika kelima program masuk dalam antrian sekaligus/tanpa waktu kedatangan (AT) dan quantum (q) = 3.

Proses	Burst Time (BT)
P1	12
P2	8
P3	5
P4	2
P5	1

Grafik *Hantt* sebagai berikut :



Dari grafik di atas, diperoleh rata-rata waktu tunggu dan rata-rata *turn around time* sebagai berikut :

$$\begin{aligned}
 P1 &= (12-3)+(20-15)+(25-23) &= 16 \\
 P2 &= (15-6)+(23-18) &= 14 \\
 P3 &= (18-9) &= 9 \\
 P4 &= 0 &= 0 \\
 P5 &= 0 &= 0 \\
 \text{Jumlah} & &= 39
 \end{aligned}$$

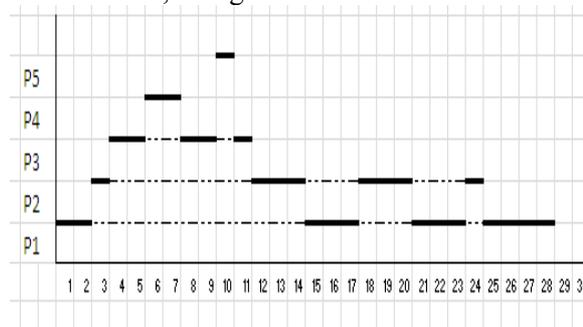
$$\begin{aligned}
 \text{AverageWaiting Time (AWT)} &= 7.8 \\
 \text{Average Turn Around Time (ATAT)} &= 13.4 \\
 \text{Performance CPU} &= 0.128
 \end{aligned}$$

- b. Eksekusi dengan Burst Time (BT) dan waktu kedatangan (AT) yang diketahui.

Artinya kelima proses tersebut meng-antri dengan waktu kedatangan diketahui dan $q = 3$.

Proses	Arival Time (AT)	Burst Time (BT)
P1	0	12
P2	2	8
P3	3	5
P4	5	2
P5	9	1

Grafik *Hantt*, sebagai berikut :



Dari grafik di atas, diperoleh rata-rata waktu tunggu dan rata-rata *turn around time* sebagai berikut :

$$\begin{aligned}
 P1 &= (14-2)+(20-17)+(24-23) &= 16 \\
 P2 &= (11-3)+(17-14) + (23-20) &= 24
 \end{aligned}$$

$$\begin{aligned}
 P3 &= (7-5) + (10-9) && = 3 \\
 P4 &= 0 && = 0 \\
 P5 &= 0 && = 0 \\
 \text{Jumlah} &&& = 43 \\
 \text{Average Waiting Time (AWT)} &= && 8.6 \\
 \text{Average Turn Around Time (ATAT)} &= && 12.2 \\
 \text{Performance CPU} &= && 0.116
 \end{aligned}$$

Dari kedua percobaan di atas terlihat bahwa pada percobaan (a), dengan 5 proses tanpa AT diperoleh rata-rata waiting time (AWT) sebesar 7.8 satuan waktu, rata-rata turn around time (ATAT) sebesar 13.4 satuan waktu dan performance CPU sebesar 0.128. Sedangkan pada percobaan (b) dengan memperhitungkan faktor AT diperoleh rata-rata waiting time (AWT) sebesar 8.6 satuan waktu, ATAT sebesar 12.2 satuan waktu dan performance CPU-nya sebesar 0.116. Jika proses yang berada dalam antrian semakin banyak, maka akan terjadi peningkatan AWT dan ATAT sehingga performance CPU akan menjadi lebih lambat.

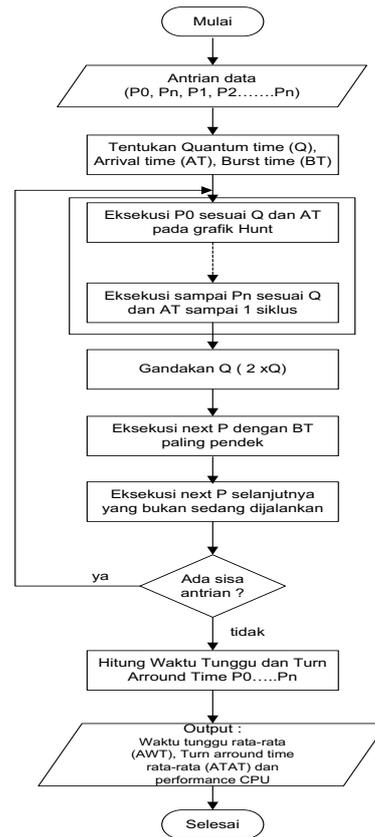
5. Pengembangan Penjadwalan Round-Robin.

Untuk mengurangi/menurunkan rata-rata waiting time (AWT) dan turn around time (ATAT) pada 5 proses program di atas, maka dibangun pengembangan penjadwalan Round Robin sebagai berikut:

Proses dimulai dengan meninjau antrian data P_1, P_2, \dots, P_n . Kemudian menentukan quantum time (q) dan menerapkan pada setiap proses melalui Grafik Huntt (*huntt chart*).

Eksekusi proses P_1, P_2, \dots, P_n dalam antrian berdasarkan q yang diberikan hingga selesai satu siklus. Jika telah selesai, gandakan quantum time menjadi $2 \cdot q$ dan eksekusi sisa proses (P_1, P_2, \dots, P_n) yang memiliki burst time (BT) paling pendek dari antrian.

Selanjutnya, pilih proses terpendek berikutnya dan eksekusi, dengan catatan bahwa proses yang dipilih ini bukan termasuk proses yang telah dijalankan sebelumnya dalam siklus yang sedang berjalan. Jika masih ada sisa antrian, maka ulangi dari awal.



Gambar 5.1. Pengembangan Round Robbin

Jika algoritma ini diterapkan pada 5 proses sebagaimana pada bagian (5), maka aktifitasnya sebagai berikut :

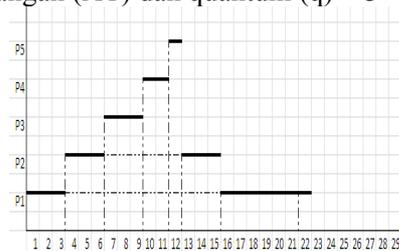
```

P1 : pushl %edi
P2 : pushl %esi
P3 : imul 16(%ebp), %eax
P4 : leal 0(,%eax,4), %ecx
P5 : movl %ebx, %edx
    
```

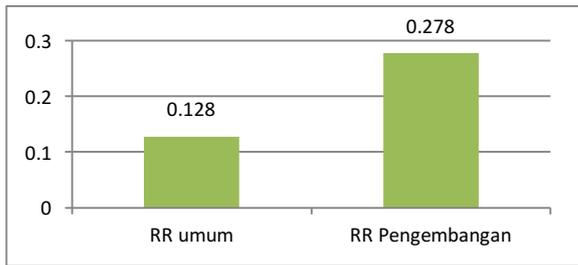
Jika diolah dengan algoritma pengembangan Round Robin sebagai berikut :

- a. Proses *real time* dengan *Burst Time* (BT) saja.

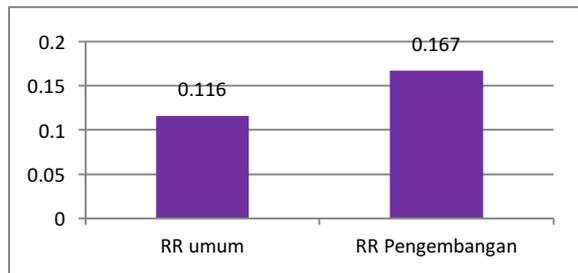
Artinya ketika kelima program masuk dalam antrian sekaligus/tanpa waktu kedatangan (AT) dan quantum (q) = 3



6. Grafik perbandingan performance CPU tanpa arrival time (AT)



7. Grafik perbandingan performance CPU menggunakan arrival time (AT)



Grafik (1) dan (2) adalah grafik AWT dan ATAT ketika *arrival time* tidak diperhitungkan baik pada RR umum dan RR pengembangan. Terlihat bahwa ketika menerapkan Round Robbin pengembangan AWT turun dari 7.8 menjadi 3.6 satuan waktu sedangkan ATAT turun dari 13.4 ke 8 satuan waktu. Grafik (3) dan (4) adalah grafik AWT dan ATAT ketika *arrival time* diperhitungkan. Terlihat bahwa ketika menerapkan Round Robbin pengembangan AWT turun dari 8.6 menjadi 6 satuan waktu sedangkan ATAT turun dari 12.2 ke 11.6 satuan waktu. Grafik (5) dan (6) adalah grafik perbandingan *performance* CPU tanpa *arrival time* dan menggunakan *arrival time*. Terlihat bahwa ketika tidak menerapkan *arrival time*, *performance* CPU meningkat dari 0.128 menjadi 0.278. Sedangkan ketika menerapkan *arrival time* *performance* CPU meningkat dari 0.116 menjadi 0.167.

Perbandingan dengan pengujian sebelumnya yang dilakukan oleh Ajit Singh, Priyanka Goyal dan Sahil Bahtra (2012) adalah sebagai berikut :

Pengujian sebelumnya	Pengujian sekarang
Menjelaskan tahapan RR pengembangan dalam 3 phase saja.	Menjelaskan algoritma lebih rinci hingga output yang mendukung <i>performance</i> CPU.

Proses diuji menggunakan <i>Gantt chart</i> .	Proses diuji menggunakan <i>Huntt chart</i> .
Parameter yang diuji adalah jumlah <i>switching</i> , AWT dan ATAT.	Parameter yang diuji adalah AWT, ATAT dan <i>Performance</i> CPU
Kesimpulan yang diambil tidak menyertakan angka pasti hasil pengujian.	Kesimpulan dengan menyertakan angka pasti pengujian (AWT, ATAT dan <i>Performance</i> CPU)

7. Kesimpulan

Kesimpulan dari laporan ini adalah :

1. Penjadwalan Round Robin pada dasarnya menggunakan *time sharing* dengan setiap proses mendapatkan waktu CPU (quantum).
2. Pengolahan proses *real time* CPU dapat dilakukan dengan menerapkan algoritma pengembangan RR. Hasil pengujian telah terjadi penurunan AWT dan ATAT yang akan mempengaruhi terjadinya peningkatan *performance* CPU.
3. Dengan menerapkan algoritma pengembangan Round Robbin, telah berhasil meningkatkan *performance* CPU ketika melakukan *fetching* data.

8. Daftar Pustaka

- [1] Randal E. Bryant, David R. O'Hallaron, *Computer System A Programmer's Perspective*, Prentice Hall, 2003, 281
- [2] Ajit Singh, Priyanka Goyal, Sahil Bahtra, *An Optimized Round Robin Sceduling algorithm for CPU sceduling*, International Journal on Computer Science and Engineering Vol. 02 No 07 Tahun, 2012
- [3] Abbas Noon, Ali Kalakech, Seifedine Kadry, *A New Round Robin Based Sceduling Algorithm for Operating System: Dinamyc Quantum using the mean average*, Faculty of Business Lebanase University.
- [4] C. Yaashuwanth, R. Ramesh, *A New Scheduling algorithm for real time task*, International Journal of computer science and invormation security, vol 6 no 2, 2009.
- [5] Abas Ali Pangera, Dony Ariyus, *Sistem Operasi*, ANDI Yogyakarta, 2010.

SISTEM KEAMANAN DAN MONITORING RUMAH PINTAR SECARA ONLINE MENGGUNAKAN PERANGKAT MOBILE

¹Zainal Abidin, ²Susmini Indriani Lestaringati

^{1,2}Jurusan Teknik Komputer, Fakultas Teknik dan Ilmu Komputer
Universitas Komputer Indonesia, Jl. Dipati Ukur No.112-116, Bandung 40132
Email : ¹jgp.uswh@gmail.com, ²susmini.indriani@email.unikom.ac.id

ABSTRAK

Sebuah rumah pintar atau lebih dikenal dengan istilah *smart home* adalah sebuah tempat tinggal atau kediaman yang menghubungkan jaringan komunikasi dengan peralatan listrik untuk dimungkinkan dikontrol, dimonitor atau diakses dari jarak jauh. Sedangkan sistem *smart home* yang ada dipasaran memiliki harga yang cukup mahal dan komunikasi antar perangkat menggunakan standar tertentu sehingga tidak mudah jika ingin dilakukan pengembangan lebih lanjut. Tujuan penelitian ini adalah dengan memanfaatkan teknologi berbasis IP (*Internet Protocol*), dengan menggunakan protokol *TCP/IP* menghubungkan semua peralatan menggunakan protokol yang sama, sehingga akan memudahkan komunikasi antar perangkat. Untuk memudahkan pengguna, perangkat untuk mengakses lingkungan rumah dapat menggunakan perangkat mobile yang terhubung dengan Internet. Pengujian akan dilakukan terhadap sebuah *prototype* rumah yang memiliki 3 buah lampu, sebuah pintu yang dilengkapi solenoid door lock untuk mewakili penguncian, dan 2 buah IP Camera. Dari hasil pengujian aplikasi yang dibangun telah dapat mengontrol peralatan rumah seperti lampu dan kunci pintu, dapat memantau keadaan dan kondisi melalui IP Camera, dan dapat memberikan peringatan ketika kunci pintu dibuka secara paksa.

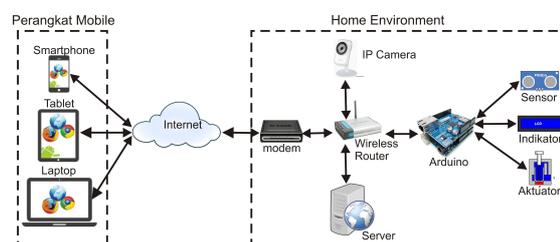
Kata kunci: Smart Home, TCP/IP, Perangkat Mobile

1. PENDAHULUAN

Sebuah rumah pintar atau lebih dikenal dengan istilah *smart home* adalah sebuah tempat tinggal atau kediaman yang menghubungkan jaringan komunikasi dengan peralatan listrik untuk dimungkinkan dikontrol, dimonitor atau diakses dari jarak jauh [1]. Tujuan dibangunnya konsep rumah pintar adalah memberikan kenyamanan kepada si pemilik rumah untuk dapat mengontrol serta memonitoring rumahnya ketika pemilik rumah sedang tidak berada dilingkungan rumah mereka. Sistem *smarthome* yang ada dipasaran cenderung membutuhkan harga yang cukup mahal serta membutuhkan instalasi yang rumit. Selain itu komunikasi antar perangkat menggunakan standar tertentu sehingga tidak mudah jika ingin dilakukan pengembangan lebih lanjut.

Sebagai pusat dari pengontrolan dapat menggunakan mikrokontroler Arduino. Arduino sendiri memiliki beberapa keunggulan seperti murah, sederhana didalam pembuatan programnya, berbasis open source serta memiliki modul siap pakai (*shield*) yang bisa ditancapkan pada *board* Arduino, misalnya Shield GPS, Ethernet, SD Card, dan lain sebagainya. Dengan memanfaatkan teknologi berbasis IP (*Internet Protocol*), dengan menggunakan protokol *TCP/IP* menghubungkan semua peralatan menggunakan protokol yang sama, sehingga akan memudahkan komunikasi antar perangkat. Untuk memudahkan pengguna, perangkat untuk mengakses lingkungan rumah dapat menggunakan perangkat mobile yang terhubung dengan Internet. Pada gambar 1.1

dibawah ini merupakan gambaran umum dari sistem *smarthome*.



Gambar 1.1 Gambaran Umum Sistem Smarthome

Pada penelitian ini akan dibangun sebuah sistem keamanan dan monitoring rumah pintar secara online menggunakan perangkat mobile, seperti smartphone, tablet maupun laptop. Harapannya sistem ini dapat membantu pemilik rumah untuk dapat menjaga rumahnya dari jarak jauh.

2. TINJAUAN PUSTAKA

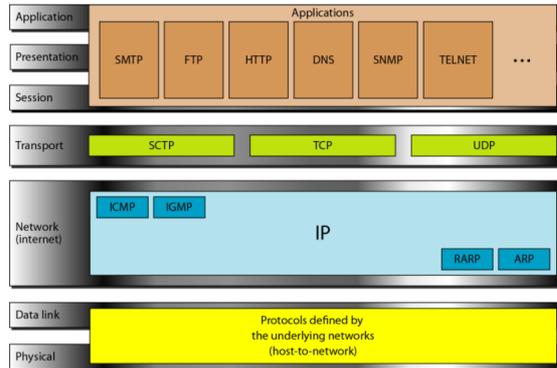
2.1. Internet

Internet (kependekan dari *interconnection-networking*) adalah seluruh jaringan komputer yang saling terhubung menggunakan standar sistem global *Transmission Control Protocol/Internet Protocol Suite* (*TCP/IP*) sebagai protokol pertukaran paket (*packet switching communication protocol*) untuk melayani miliaran pengguna di seluruh dunia. Jaringan Internet pada saat ini merupakan jaringan yang sangat kompleks dan memiliki struktur yang dinamis (selalu

berkembang). Internet tersusun atas banyak jaringan WAN dan LAN. Para pemakai akhir (*end user*) dapat berhubungan dengan jaringan Internet melalui sebuah Internet Service Provider (ISP). [2]

2.2. TCP/IP Protocol Suite

Layer pada protokol TCP/IP tidak sama persis dengan model OSI. Model TCP/IP memiliki empat layer, yaitu Host to network, Internet, Transport, dan Application. Gambar dibawah merupakan gambar dari susunan Model TCP/IP.



Gambar 2.1 TCP/IP Protocol Suite

Tiap layer pada gambar diatas dijelaskan sebagai berikut:

- Layer Host-to-Network (Host-to-Network Layer)**
Layer ini merupakan gabungan dari layer fisik dan layer jalur data pada layer model OSI. Terdiri dari protokol-protokol dasar jaringan seperti ethernet, token ring, frame relay, dan ATM.
- Layer Internet (Internet Layer)**
Fungsinya sama dengan layer jaringan pada model OSI. Protokol yang bekerja pada layer ini diantaranya adalah Internet Protokol (IP), Internet Control Message Protocol (ICMP), Internet Group Message Protocol (IGMP), Address Resolution Protocol (ARP), Reverse Address Resolution Protocol (RARP).
- Layer Transport (Transport Layer)**
Secara tradisional layer ini memiliki dua protokol, yaitu Transmission Control Protocol (TCP) dan User Datagram Protocol (UDP).
- Layer Aplikasi (Application Layer)**
Layer aplikasi pada model TCP/IP merupakan gabungan dari layer sesi, presentasi, dan aplikasi pada layer model OSI. Pada layer ini banyak protokol yang digunakan, diantaranya adalah Simple Mail Transfer Protocol (SMTP), HyperText Transfer Protocol (HTTP), dan lain sebagainya.[3]

2.3. Arduino

Arduino adalah kit elektronik atau papan rangkaian elektronik *open source* yang didalamnya terdapat komponen utama, yaitu sebuah chip

mikrokontroler dengan jenis AVR dari perusahaan Atmel. Mikrokontroler itu sendiri adalah chip atau IC (*Integrated Circuit*) yang bisa diprogram menggunakan komputer. Tujuan memprogram mikrokontroler adalah agar rangkaian elektronika dapat membaca *input*, memproses *input* tersebut, dan kemudian menghasilkan *output* sesuai yang diinginkan. Jadi mikrokontroler bertugas sebagai otak yang mengendalikan *input*, *proses*, dan *output* pada sebuah rangkaian elektronik.

Secara umum, Arduino terdiri dari dua bagian, yaitu *hardware* dan *software*. Hardware merupakan berupa papan *input/output* yang *open source*. Sedangkan *software* berupa Software Arduino yang juga *open source*, meliputi software IDE untuk menulis program dan *driver* untuk koneksi dengan komputer.

Kelebihan dari Arduino dari *platform* lain yaitu:

- Murah, Arduino biasanya dijual relatif murah dibandingkan dengan *platform* mikrokontroler lainnya.
- Sederhana dan mudah pemrogramannya. Perlu diketahui bahwa lingkungan pemrograman di arduino mudah digunakan dan cukup fleksibel bagi. Perangkat lunaknya *open source*. Perangkat lunak Arduino IDE dipublikasikan sebagai *open source*.
- Perangkat kerasnya *open source*. Perangkat kerasnya dipublikasikan *open source*. Perangkat keras Arduino berbasis mikrokontroler ATMEGA168, ATMEGA328, dan ATMEGA1280.
- Tidak perlu perangkat *chip programmer*. Karena di dalamnya sudah ada *bootloader* yang akan menangani *upload* program dari komputer.
- Sudah memiliki komunikasi USB. Sehingga perangkat Laptop yang tidak memiliki port serial/RS323 bisa menggunakannya.
- Bahasa pemrograman relative mudah. Karena *software* Arduino dilengkapi dengan kumpulan *library* yang cukup lengkap.
- Memiliki modul siap pakai (*shield*) yang bisa ditancapkan pada *board* Arduino. Misalnya Shield GPS, Ethernet, SD Card, dan lain-lain. [4]

2.4. Perangkat Lunak Pendukung

Untuk membangun sistem monitoring dibutuhkan program aplikasi atau perangkat lunak yang bisa membantu dalam membangun sistem monitoring.

2.4.1. Visual Basic 6.0

Visual Basic 6.0 adalah *development tools* yang fleksibel dengan menggunakan model pemrograman (COM). Bahasanya lebih mudah dimengerti dari pada bahasa lain, begitu juga dengan IDE (*Intergrated Development*

Sistem Keamanan Dan Monitoring Rumah Pintar Secara Online Menggunakan Perangkat Mobile

Environment) pada Visual Basic 6.0 yang *User Friendly* atau mudah dipahami oleh pengguna sehingga orang awam pun bisa menggunakannya.

Visual Basic merupakan turunan dari bahasa pemrograman BASIC dan menawarkan pengembangan perangkat lunak komputer berbasis grafik dengan cepat. Beberapa bahasa skrip seperti Visual Basic for Applications (VBA) dan Visual Basic Scripting Edition (VBScript), mirip seperti halnya Visual Basic, tetapi cara kerjanya yang berbeda. Para programmer dapat membangun aplikasi dengan menggunakan komponen yang disediakan oleh Microsoft Visual Basic. Program-program yang ditulis dengan Visual Basic juga dapat menggunakan Windows API (*Application Programming Interface*), tetapi membutuhkan deklarasi fungsi luar tambahan.[5]

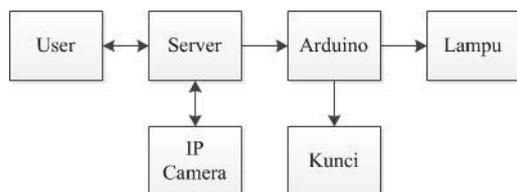
2.4.2. Arduino IDE

Arduino development berisi editor teks untuk kode, area pesan, konsol teks, toolbar dengan tombol untuk fungsi-fungsi umum menulis, dan serangkaian menu. Ini menghubungkan ke perangkat keras Arduino untuk meng-upload program dan untuk berkomunikasi.

Software yang ditulis menggunakan Arduino disebut sketsa. Sketsa ini ditulis dalam editor teks. Sketsa disimpan dengan ekstensi file .ino. Ini memiliki fitur untuk meng-cut/paste dan untuk mencari / mengganti teks. Daerah pesan memberikan umpan balik sambil menyimpan dan mengekspor serta menampilkan kesalahan. Konsol menampilkan teks output dengan lingkungan Arduino termasuk pesan error lengkap dan informasi lainnya. Bagian bawah sebelah kanan sudut jendela menampilkan papan saat ini dan port serial. Tombol-tombol toolbar memungkinkan Anda untuk memverifikasi dan mengunggah program, membuat, membuka, dan menyimpan sketsa, dan membuka monitor serial. [6]

3. PERANCANGAN

Langkah awal dari perancangan sistem adalah analisis dan penentuan kebutuhan sistem. Pada langkah ini menentukan kebutuhan apa saja yang harus dipenuhi. Pada gambar 3.1 adalah diagram blok sistem yang akan dibangun.

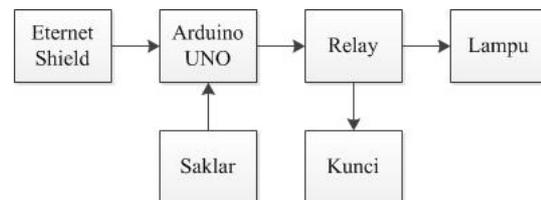


Gambar 3.1 Diagram Blok Sistem Monitoring dan Kontrol fungsi Rumah

Gambar 3.1 merupakan diagram blok sistem dengan penjelasan masing – masing blok sebagai berikut :

1. User adalah antarmuka sistem berupa PC ataupun Smartphone sebagai media akses secara *online*.
2. Server bertugas sebagai perangkat yang digunakan sebagai antarmuka sistem secara jaringan lokal (*offline*) dan dapat menerima perintah dari user. Server juga yang memberikan perintah – perintah ke IP Camera dan Arduino melalui jaringan lokal.
3. IP Camera bertugas sebagai perangkat yang memonitoring ruang.
4. Arduino bertugas mengontrol lampu dan kunci sesuai dari yang diperintahkan oleh server.
5. Kunci adalah perangkat yang akan dikontrol.
6. Lampu adalah perangkat yang akan dikontrol.

Arduino UNO tidak dapat berkomunikasi langsung dengan Server melalui jaringan lokal dan tidak dapat berkomunikasi langsung dengan lampu dan kunci. Maka Arduino UNO harus ditambahkan dengan Ethernet Shield untuk dapat berkomunikasi dengan Server dan Relay agar dapat memerintahkan lampu dan kunci. Gambar 3.2 adalah gambaran susunan diagram blok Arduino UNO dengan Ethernet Shield dan Relay.



Gambar 3.2 Diagram Blok Arduino dengan Ethernet Shield dan Relay

Gambar 3.2 adalah gambaran susunan diagram blok Arduino dengan Ethernet Shield dan Relay dengan penjelasan masing – masing bloknya sebagai berikut:

1. Ethernet Shield adalah perangkat tambahan pada Arduino UNO agar Arduino Uno dapat berkomunikasi dengan Server melalui jaringan lokal.
2. Arduino UNO bertugas mengontrol Lampu dan kunci melalui relay sesuai perintah dari Server dan perintah dari saklar.
3. Saklar adalah perangkat antarmuka manual untuk mengontrol lampu.
4. Relay adalah perangkat yang bertugas memerintahkan lampu dan kunci sesuai apa yang diperintahkan Arduino UNO.
5. Kunci adalah perangkat yang akan dikontrol.
6. Lampu adalah perangkat yang akan dikontrol.

3.1. Komponen Sistem

3.1.1. Perangkat Keras

Pada bagian ini menjelaskan kebutuhan perangkat keras untuk membangun sistem monitoring dan kontrol fungsi rumah, tabel 3.1 berikut daftar perangkat keras.

Tabel 3.1 Perangkat Keras

No	Perangkat Keras	Jumlah
1	Laptop	1 buah
2	IP Camera D-Link DSC-933L	1 buah
2	IP Camera D-Link DSC-5020L	1 buah
3	Arduino Uno	1 buah
4	Arduino Ethernet Shield	1 buah
5	Router-wifi	1 buah
6	Solenid Door Lock	1 buah
7	Kabel UTP	2 buah x 3meter
8	Adaptor Arduino UNO	1 buah
9	Relay HRS DC5V	6 buah
10	Lampu	3 buah
11	Saklar	3 buah

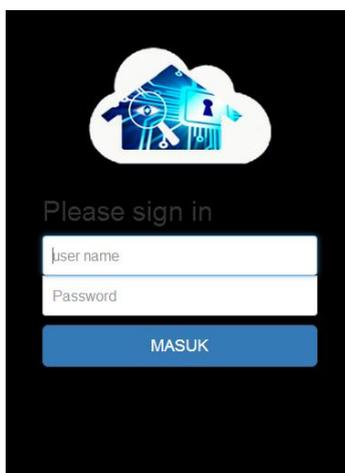
3.1.2. Perangkat Lunak

Terdapat dua buah perangkat lunak yang dibutuhkan yaitu IDE Arduino 1.6.2 sebagai editor untuk Arduino serta Visual Basic 6 sebagai editor untuk antarmuka pada server.

4. PENGUJIAN DAN ANALISA

4.1. Pengujian Antar Muka

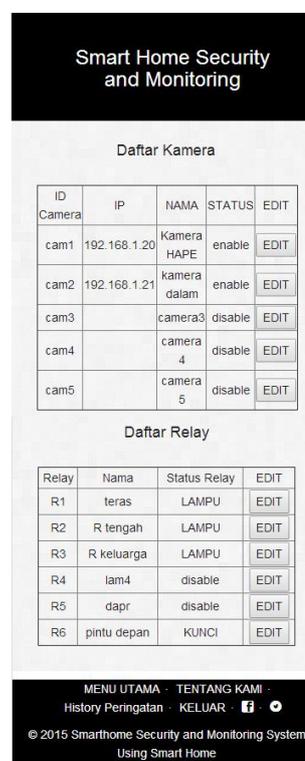
Berikut ini adalah pengujian antarmuka menggunakan smartphone asusu zenfone 5. Halaman antarmuka terdiri dari halaman login, halaman utama dan halaman untuk pengaturan.



Gambar 4.1 Tampilan Halaman Login Pengguna Menggunakan Smartphone



Gambar 4.2 Tampilan Halaman Utama Menggunakan Smartphone



Gambar 4.3 Tampilan Halaman Pengaturan Daftar Kamera dan Daftar Relay

Sistem Keamanan Dan Monitoring Rumah Pintar Secara Online Menggunakan Perangkat Mobile

4.2. Pengujian sistem pada Perangkat

Berikut ini adalah hasil pengujian pengontrolan alat menggunakan perangkat Asus Zenfone 5, dengan web browser Google Chrome.

Tabel 4.1 Pengujian Pengontrolan

No	Jenis Peralatan	Kondisi	
		On	Off
A	Pada Smartphone		
1	Lampu 1	Berhasil	Berhasil
2	Lampu 2	Berhasil	Berhasil
3	Lampu 3	Berhasil	Berhasil
4	Kunci 1	Berhasil	Berhasil
B	Pada Tablet		
1	Lampu 1	Berhasil	Berhasil
2	Lampu 2	Berhasil	Berhasil
3	Lampu 3	Berhasil	Berhasil
4	Kunci 1	Berhasil	Berhasil
C	Pada Laptop		
1	Lampu 1	Berhasil	Berhasil
2	Lampu 2	Berhasil	Berhasil
3	Lampu 3	Berhasil	Berhasil
4	Kunci 1	Berhasil	Berhasil

Pengujian streaming kamera 1 dan kamera 2 pada perangkat asus zenfone 5.

Tabel 4.2 Pengujian streaming kamera

No	Jenis Peralatan	Pengujian Streaming
A	Pada Smartphone	
1	Kamera 1	Berhasil
2	Kamera 2	Berhasil
B	Pada Tablet	
1	Kamera 1	Berhasil
2	Kamera 2	Berhasil
C	Pada Laptop	
1	Kamera 1	Berhasil
2	Kamera 2	Berhasil

Pengujian peringatan ketika ada yang membuka paksa kunci pintu pada perangkat asus zenfone 5.

Tabel 4.3 Pengujian Peringatan/Pemberitahuan

No	Kondisi Pintu	Kondisi Kunci	Peringatan
A	Pada Smartphone		
1	Terbuka	Terkunci	Ada
2	Tertutup	Terkunci	Tidak Ada
3	Terbuka	Tidak Terkunci	Tidak Ada
4	Tertutup	Tidak Terkunci	Tidak Ada
B	Pada		

No	Kondisi Pintu	Kondisi Kunci	Peringatan
	Tablet		
1	Terbuka	Terkunci	Ada
2	Tertutup	Terkunci	Tidak Ada
3	Terbuka	Tidak Terkunci	Tidak Ada
4	Tertutup	Tidak Terkunci	Tidak Ada
C	Pada Laptop		
1	Terbuka	Terkunci	Ada
2	Tertutup	Terkunci	Tidak Ada
3	Terbuka	Tidak Terkunci	Tidak Ada
4	Tertutup	Tidak Terkunci	Tidak Ada

5. KESIMPULAN DAN SARAN

Dari hasil pengujian yang telah dilakukan, kesimpulan yang dapat diambil adalah sebagaimana berikut:

1. Aplikasi yang dibangun telah dapat pengontrolan peralatan rumah seperti lampu dan kunci pintu.
2. Aplikasi yang dibangun telah dapat memantau (memonitoring) keadaan dan kondisi melalui *IP Camera*.
3. Aplikasi yang dibangun telah dapat memberikan peringatan ketika kunci pintu dibuka secara paksa.

Saran untuk pengembangan kedepan adalah sistem keamanan yang dibangun diperlukan tambahan keamanan seperti motion detection serta menambahkan pemberitahuan kepada pengguna dalam berupa sms atau email kepada pemilik rumah.

DAFTAR PUSTAKA

- [1] Grabowski, Mateusz. & Grzegorz Dziwoki. (2009). *The IEEE Wireless Standards as an Infrastructure of Smart Home Network Computer Networks*, Volume 39, pp 302-309
- [2] Irawan, Budhi. (2005). *Jaringan Komputer*. Yogyakarta: Graha Ilmu.
- [3] Forouzan, Behrouz A. (2007). *TCP/IP Protocol Suite (Vol 4)*. New York: Mc Graw-Hill.
- [4] Syahwil, Muhammad. (2013). *Mikrokontroler Arduino*. Yogyakarta: Andi.
- [5] Abdul, Razaq. (2004). *Pemrograman Microsoft Visual Basic*. Surabaya: Indah.
- [6] <https://www.arduino.cc/en/guide/environment>. Diakses 02 juli 2015

SISTEM MONITORING DISTRIBUSI BANTUAN BENCANA ALAM BERBASIS WEB MENGUNAKAN METODE ALGORITMA FIRST-FIT

Sri Nurhayati¹, Ryan Zulmi²

^{1,2}Jurusan Sistem Komputer Universitas Komputer Indonesia, Bandung

¹serieid@yahoo.com, ²ryanzulmi@gmail.com

ABSTRAK

Ketidakterataan bantuan yang didistribusikan kepada korban bencana di posko pengungsian disebabkan oleh donatur dari instansi atau perusahaan melakukan pemberian bantuan secara langsung kepada korban di posko pengungsian yang mengakibatkan penumpukan bantuan di satu titik. Untuk itu, pada penelitian ini dibuat sistem monitoring distribusi bantuan bencana alam berbasis website menggunakan metode algoritma first-fit yang dapat meminimalisasi terjadinya ketidakmerataan tersebut. Algoritma first fit akan menentukan kemana bantuan disalurkan kepada posko pengungsian yang membutuhkan sesuai dengan besarnya jumlah kebutuhan logistik sehari-hari dengan tepat. Selain itu, sistem ini dibantu dengan penggunaan barcode scanner yang berfungsi untuk memudahkan proses verifikasi barang bantuan yang telah sampai di posko pengungsian, yaitu dengan cara men-scan barcode yang tertera pada barang dan secara otomatis data barang terverifikasi masuk ke basis data sistem. Model sistem dirancang menggunakan pendekatan orientasi objek dengan salah satu toolsnya yaitu UML. Dari hasil pengujian, sistem monitoring distribusi bantuan bencana alam menggunakan metode algoritma first-fit dapat membantu Badan Penanggulangan Bencana Daerah dalam menentukan posko pengungsian mana yang tepat diberikan materi bantuan.

Kata kunci : Bencana Alam, Distribusi Bantuan, Algoritma First-fit

1. PENDAHULUAN

Bencana alam adalah suatu peristiwa alam yang mengakibatkan dampak besar bagi populasi manusia. Banyak korban bencana yang membutuhkan pertolongan untuk bisa bertahan hidup dari bencana alam. Badan Penanggulangan Bencana Daerah (BPBD) merupakan lembaga pemerintah non-departemen yang melaksanakan tugas penanggulangan bencana. Salah satu masalah yang muncul ketika terjadi bencana alam adalah kurangnya koordinasi antara para donatur dengan badan penanggulangan bencana dalam hal pendistribusian bantuan logistik. Beberapa donatur dari instansi atau perusahaan melakukan pemberian bantuan secara langsung kepada korban di lokasi bencana. Hal ini dapat menyebabkan ketidakmerataan distribusi bantuan. Masalah lainnya, lemahnya pemantauan pendistribusian bantuan kepada korban bencana atau penerima bantuan yang dilakukan oleh Badan Penanggulangan Bencana Daerah (BPBD). Hal ini membuat banyak korban bencana yang merasa dirinya belum menerima bantuan padahal bantuan sudah didistribusikan dan diterima oleh perwakilan posko pengungsian (penerima) tersebut. Masalah lainnya adalah materi bantuan terutama bantuan logistik yang telah diperkirakan oleh Badan Penanggulangan Bencana Daerah untuk dipakai perhari maupun perminggu, terkadang secara tiba-tiba tidak mencukupi kebutuhan tersebut. Hal ini disebabkan oleh adanya korban bencana palsu yang berpura-pura menjadi korban di pengungsian-pengungsian yang ada.

Dari masalah diatas, maka peneliti akan membangun sebuah sistem monitoring distribusi bantuan bencana alam berbasis website dengan menggunakan metode algoritma first-fit. Algoritma

first-fit merupakan algoritma manajemen memori yang biasa dipergunakan oleh sistem operasi pada komputer untuk mengalokasikan proses-proses ke dalam memori kosong yang tersedia. Algoritma first-fit dapat diimplementasikan pada sistem pendistribusian materi bantuan yaitu, akan mengalokasikan materi bantuan ke posko-posko pengungsian dengan tepat.

2. TEORI PENUNJANG

2.1 Sistem Distribusi

Pendistribusian bantuan merupakan mekanisme atau prosedur yang menghubungkan antara pemberi bantuan (donator) dan penerima bantuan (korban bencana) pada suatu situasi kebencanaan. Bantuan dalam hal ini adalah bantuan kemanusiaan yang terdiri dari tempat penampungan sementara, bantuan pangan, sandang, air bersih dan sanitasi, serta pelayanan kesehatan.

Dalam pelaksanaan pendistribusian atau pemberian bantuan terdapat beberapa langkah, diantaranya:

1. Penyusunan Daftar Penerima Bantuan
Langkah-langkah yang dilakukan pada kegiatan pendaftaran penerima bantuan adalah sebagai berikut:
 - a) Mengidentifikasi jumlah keluarga di setiap titik penampungan.
 - b) Berdasarkan identifikasi jumlah keluarga, disusun daftar penerima bantuan secara keseluruhan sesuai dengan kelompok umur.
2. Penilaian Kebutuhan
Langkah-langkah yang dilakukan pada kegiatan penilaian kebutuhan adalah sebagai berikut:
 - a) Mengidentifikasi kebutuhan penerima bantuan (korban bencana).

Sistem Monitoring Distribusi Bantuan Bencana Alam Berbasis Website Menggunakan Metode Algoritma First-Fit

- b) Mengidentifikasi sumber barang-barang/aset yang masih dimiliki korban/penerima bantuan.
 - c) Menentukan jenis bantuan yang diperlukan penerima bantuan (korban bencana).
3. Penilaian Jenis Bantuan
- Langkah berikutnya menentukan jumlah bantuan yang harus didistribusikan pada seluruh penerima bantuan (korban bencana). langkah-langkah:
- a) Menyiapkan daftar penerima bantuan dan daftar kebutuhan yang diperlukan sesuai dengan prioritas.
 - b) Menghitung perkiraan jumlah yang harus didistribusikan pada setiap hari/minggu/bulan.
 - c) Menghitung jumlah bantuan tidak terduga.
 - d) Menghitung jumlah keseluruhan bantuan yang diperlukan selama masa di penampungan.
4. Pendistribusian Bantuan
- Pendistribusian bantuan harus cepat dan tepat serta sesuai dengan kondisi setempat, dengan melalui beberapa mekanisme:
- a) Penerima bantuan pangan diidentifikasi dan menjadi sasaran berdasarkan kebutuhan.
 - b) Metode distribusi dirancang melalui konsultasi dengan kelompok setempat, lembaga-lembaga mitra, dan melibatkan berbagai kelompok penerima.
 - c) Titik-titik distribusi sedekat mungkin dengan hunian sementara penerima untuk memastikan akses yang mudah dan aman.
 - d) Kualitas, jumlah jatah makanan/pangan dan rencana distribusi diinformasikan jauh sebelumnya kepada penerima bantuan.
 - e) Kinerja dan efektifitas program bantuan pangan dimonitor dan dievaluasi dengan semestinya.

2.2 Bencana

Bencana adalah peristiwa atau rangkaian peristiwa yang mengancam dan mengganggu kehidupan dan penghidupan masyarakat yang disebabkan, baik oleh faktor alam dan/atau faktor nonalam maupun faktor manusia sehingga mengakibatkan timbulnya korban jiwa manusia, kerusakan lingkungan, kerugian harta benda, dan dampak psikologis.

2.3 Algoritma First-Fit

Algoritma first fit yaitu manajer memori mencari (men-*scan*) sepanjang list segmen sampai mendapatkan rongga atau tempat yang cukup besar. Rongga tersebut dibagi ke dalam dua bagian, satu untuk proses dan satu lagi untuk ruang kosong (*unused memory*), kecuali jika proses memerlukan seluruh rongga maka rongga tersebut hanya dipakai untuk proses. Dalam sistem distribusi ini, Algoritma first fit akan menentukan kemana

bantuan dari donator disalurkan kepada posko pengungsian yang membutuhkan sesuai dengan besarnya jumlah materi bantuan, jumlah korban di setiap posko, dan jumlah kebutuhan materi sehari-hari.

2.5 UML

UML (*Unified Modeling Language*) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. Diagram yang akan digunakan dalam pembangunan sistem monitoring distribusi bantuan berbasis website ini, yaitu *use case diagram*, *class diagram*, *sequence diagram*, dan *Activity diagram*.

2.7 Barcode

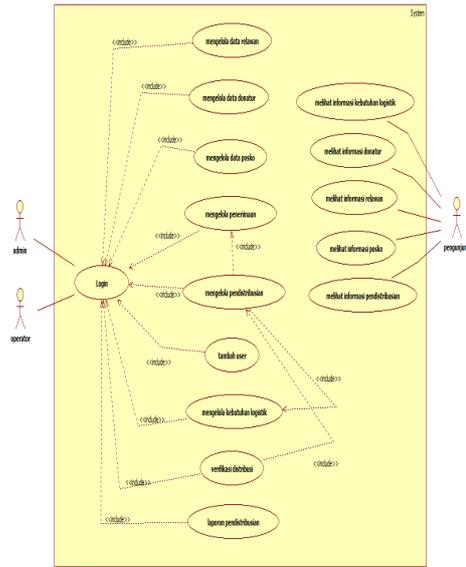
Sebuah kode batang atau kode palang (bahasa Inggris: barcode) adalah suatu kumpulan data optik yang dibaca mesin. Kode batang ini mengumpulkan data dalam lebar (garis) dan spasi garis paralel dan dapat disebut sebagai kode batang atau simbologi linear atau 1D (1 dimensi). Barcode menyimpan data-data spesifik seperti kode produksi, nomor identitas, dan lain-lain sehingga sistem komputer dapat mengidentifikasi dengan mudah informasi yang dikodekan dalam barcode.

Komputer tidak secara langsung dapat membaca data yang terkandung dalam kode bar tersebut, oleh karena itu sebelumnya kode yang ada harus ditangkap dan diterjemahkan ke dalam format data yang dapat dibaca oleh komputer. Alat yang dapat membaca dan mengirimkannya ke dalam komputer itulah yang disebut Barcode Reader atau yang biasa disebut Barcode Scanner. Seperangkat Barcode Scanner terdiri dari scanner, decoder dan kabel yang menyambungkan decoder dengan komputer. Barcode Scanner tersebut memindai symbol, menangkap dan merubah kode bar menjadi data elektrik lalu mengirimkannya ke komputer dengan format data yang sederhana.

3. PERANCANGAN SISTEM

3.3 Use Case Diagram

Use case diagram merupakan suatu diagram yang menggambarkan fungsionalitas yang dimiliki oleh suatu sistem beserta aktor-aktor yang terlibat di dalamnya. Berikut *Use Case diagram* sistem monitoring distribusi bantuan bencana alam berbasis website.



Gambar 3.5 Use Case Diagram Sistem Monitoring Distribusi Bencana Alam Berbasis Website

3.4 Class Diagram

Class diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas yang ada dalam perangkat lunak yang sedang dikembangkan. Class diagram menggambarkan struktur dan deskripsi kelas beserta hubungan kelas yang satu dengan kelas yang lain.



Gambar 3.6 Class Diagram

4. PENGUJIAN SISTEM

4.1 Pengujian Alpha

Pengujian alpha dilakukan dengan menggunakan metode *Blackbox*. Pengujian *Blackbox* berfokus pada persyaratan fungsional perangkat lunak untuk melihat apakah program aplikasi menghasilkan output yang diinginkan dan sesuai dengan fungsi dari program tersebut. Apabila dari input yang diberikan proses menghasilkan

output yang sesuai dengan kebutuhan fungsionalnya, maka program aplikasi yang bersangkutan telah benar. Tetapi jika output yang dihasilkan tidak sesuai dengan kebutuhan fungsionalnya, maka masih terdapat kesalahan pada program. Rencana pengujian yang akan dilakukan pada sistem ini terlihat pada tabel 4.1.

Tabel 4.1 Rencana Pengujian Sistem

Kelas Uji	Butir Uji	Jenis Pengujian
Login Admin dan Operator	Isi data Login	Black Box
	Validasi Username dan Password	
Pengelolaan Data Relawan	Tambah Data Relawan	Black Box
	Ubah Data Relawan	
	Hapus Data Relawan	
Pengelolaan Data Donatur	Tambah Data Donatur	Black Box
	Ubah Data Donatur	
	Hapus Data Donatur	
Pengelolaan Data Posko	Tambah Data Posko	Black Box
	Ubah Data Posko	
	Hapus Data Posko	
Pengelolaan Data Penerimaan	Tambah Stok Logistik	Black Box
	Pembuatan Barcode	
	Ubah Stok Logistik	
	Hapus Stok Logistik	
Pengelolaan Data Pendistribusian	Cek Algoritma First-fit	Black Box
	Data Fix Pendistribusian	
Pengelolaan Data User	Tambah Data User	Black Box
	Ubah Data User	
	Hapus Data User	
Laporan Pendistribusian	Ubah Data Pendistribusian	Black Box
	Hapus Data Pendistribusian	
	Ubah Data ke file Pdf	
Pengelolaan Data Kebutuhan Logistik	Tambah Data Kebutuhan Logistik	Black Box
	Ubah Data Kebutuhan Logistik	
	Hapus Data Kebutuhan Logistik	

Sistem Monitoring Distribusi Bantuan Bencana Alam Berbasis Website Menggunakan Metode Algoritma First-Fit

Verifikasi Distribusi	Ubah Data Status Pendistribusian	Black Box
Logout Admin dan Operator	Keluar Dari Halaman Admin dan Operator	Black Box
Home	Menampilkan Halaman Utama Website	Black Box
Data Relawan	Menampilkan Halaman Informasi Data Relawan	Black Box
	Cari Data Relawan	
Data Donatur	Menampilkan Halaman Informasi Data Donatur	Black Box
	Cari Data Donatur	
Data Posko	Menampilkan Halaman Informasi Data Posko	Black Box
Kebutuhan Logistik	Menampilkan Halaman Informasi Kebutuhan Logistik	Black Box
Pendistribusian	Menampilkan Halaman Informasi Pendistribusian Logistik	Black Box

Berdasarkan hasil pengujian sistem yang telah dilakukan, secara keseluruhan memberikan kesimpulan bahwa proses pada sistem ini telah melalui tahap perbaikan dan sudah dimaksimalkan terhadap proses-proses tersebut dan secara fungsional sistem sudah dapat digunakan dan menghasilkan keluaran yang diharapkan.

4.2 Pengujian Beta

Pengujian beta dilakukan dengan tujuan untuk mengetahui sejauh mana kualitas dari perangkat lunak yang dibangun, apakah sudah sesuai dengan harapan atau belum. Pengujian beta dilakukan dengan cara melakukan wawancara dan kuisioner terhadap pengguna aplikasi. Berdasarkan hasil dari pengujian beta, maka dapat disimpulkan penerapan sistem monitoring distribusi bencana alam berbasis website dengan menggunakan metode algoritma first-fit sudah sesuai dengan tujuan yaitu dapat membantu Badan Penanggulangan Bencana Daerah dalam menentukan posko pengungsian mana yang tepat diberikan materi bantuan, dapat mengelola data kebutuhan korban bencana alam, dapat memberikan informasi data kebutuhan logistik korban bencana alam kepada publik, serta dapat memberikan informasi lokasi posko bantuan bencana alam.

1. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat ditarik kesimpulan :

1. Sistem monitoring distribusi bantuan bencana alam menggunakan metode algoritma first-fit sudah dapat membantu Badan Penanggulangan Bencana Daerah dalam menentukan posko pengungsian mana yang tepat diberikan materi bantuan.
2. Dari hasil pengujian, Sebesar 83,33% jawaban setuju dari pengguna bahwa sistem berbasis website ini dapat memberikan informasi data kebutuhan logistik korban bencana alam kepada publik. Sebesar 80% jawaban setuju dari pengguna bahwa sistem berbasis website ini dapat memberikan informasi lokasi posko bantuan bencana alam kepada publik. Sebesar 80% jawaban setuju dari pengguna bahwa sistem berbasis website ini dapat memberikan informasi tentang status pendistribusian bantuan logistik.
3. Sistem monitoring distribusi bantuan bencana alam menggunakan metode algoritma first-fit cocok digunakan untuk pengelolaan distribusi dengan jumlah bantuan logistik yang banyak.

5.2 Saran

Berdasarkan hasil kesimpulan, maka saran yang dapat dikemukakan agar menjadi bahan perbaikan dan pertimbangan adalah sebagai berikut :

1. Dapat dilakukan penambahan fitur GIS (*Geographic Information System*) untuk memberikan informasi lokasi posko-posko bencana.
2. Untuk fitur barcode yang digunakan pada sistem ini masih sederhana, diharapkan dapat dikembangkan lagi dengan fitur yang lebih efisien.

DAFTAR PUSTAKA

- [1] Peraturan Kepala Badan Nasional Penanggulangan Bencana Nomor 7 Tahun 2008. "Pedoman Tata Cara Pemberian Bantuan Pemenuhan Kebutuhan Dasar"
- [2] _____, Pengertian Bencana Alam dan Klasifikasi Bencana Alam, diakses pada tanggal 2 Oktober 2014, dari world wide web:
- [3] <http://jurnalapapun.blogspot.com/2014/03/pengertian-dan-klasifikasi-bencana-alam.html/>.
- [4] S. Tanenbaum, A. 2008. *Modern Operating System*. New Jersey: Pearson Prentice Hall. Hal 200.
- [5] Stallings, W. 2012. *Operating System, Internal and Design Principles*, 7th edition. United States : Pearson. Hal 336.
- [6] _____, Pengertian Website dan Sejarah Website, diakses pada tanggal 2 Oktober 2014, dari world wide web: <http://shareilmu->

- komputer.blogspot.com/2013/03/sejarah-website-dan-teknologi-terbaru.html/.
- [7] Ariani, S., dan Shalahuddin, M., 2013. *Rekayasa Perangkat Lunak terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- [8] _____, Pengertian PHP, diakses pada tanggal 2 Oktober 2014, dari worldwide web: <http://id.wikipedia.org/wiki/PHP>
- [9] _____, Pengertian Barcode, diakses pada tanggal 20 Januari 2015, dari world wide web: <http://id.wikipedia.org/wiki/Kodebatan>
- [10] _____, Pengertian Barcode Scanner, diakses pada tanggal 20 Januari 2015, dari world wide web: <http://www.axopos.com/article/apa-itubarcodescanner--69.html#.VL-aPtKUe3I>

DETEKSI LOCAL TAMPERING PADA VIDEO MENGGUNAKAN ALGORITMA LBP-TOP

Febryanti Sthevanie

Program Studi S1 Teknik Informatika, Fakultas Informatika
Universitas Telkom, Jl. Telekomunikasi Nomor 1 Bandung
sthevanie@telkomuniversity.ac.id

ABSTRAK

Video sering dijadikan sebagai bukti otentik dalam sebuah kasus persidangan. Hal ini menimbulkan permasalahan bagaimana memastikan sebuah video asli atau hasil rekayasa. Kegiatan untuk menganalisis keaslian sebuah video adalah salah satu fungsi dari video forensic. Salah satu bentuk rekayasa yang dapat dilakukan pada video adalah local tampering. Metode yang pernah digunakan untuk mendeteksi local tampering adalah cross correlation. Metode ini melakukan proses pengecekan per pixel, sehingga kurang efisien dari sisi kompleksitas waktu. Dalam penelitian ini, diusulkan sebuah metode LBP-TOP untuk melakukan proses deteksi local tampering pada video. Metode ini dipilih karena dapat melakukan ekstraksi ciri pada domain spasial dan domain temporal secara sekaligus sehingga lebih efisien dibandingkan metode cross correlation. Metode LBP-TOP ini akan digunakan untuk melakukan ekstraksi ciri pada macroblock sebuah video lalu membandingkannya pada macroblock lain dalam video yang sama. Berdasarkan hasil pengujian yang dilakukan menunjukkan bahwa kinerja LBP-TOP dalam mendeteksi local tampering lebih efisien dibandingkan metode cross correlation. Hal ini dapat terlihat dari kompleksitas waktu yang disajikan pada data video yang sama menggunakan kedua metode tersebut.

Kata Kunci : Local tampering, LBP-TOP, Cross Correlation, Video Forensic

1. PENDAHULUAN

Pada masa sekarang, aplikasi web telah menjadi media untuk berbagi informasi dalam bentuk apapun. Sudah banyak aplikasi web yang menyediakan fasilitas untuk unggah dan unduh content berupa gambar, suara dan juga video. Hal ini menimbulkan peningkatan yang luar biasa terhadap ketersediaan content multimedia yang dapat diakses oleh siapapun. Namun, hal ini tidak diimbangi dengan adanya sistem yang dapat memverifikasi apakah content multimedia yang diunggah autentik atau tidak. Karena hal ini, pada saat seseorang menjelajah dalam dunia maya, akan sangat mungkin untuk mendapatkan content multimedia yang dimanipulasi. Dalam banyak kasus, hasil manipulasi terhadap content multimedia tersebut bisa terlihat sangat nyata. Bahkan saat ini, di media sosial banyak beredar informasi berupa gambar dan video yang merupakan hasil manipulasi dan menipu banyak pengguna media sosial. Hal ini tentu saja dapat mengakibatkan masalah yang serius.

Untuk mengatasi masalah tersebut, penelitian terkait multimedia forensic banyak dilakukan [1]. Banyak teknik yang dibuat untuk mendeteksi apakah suatu content multimedia dimanipulasi atau tidak. Penelitian multimedia forensic banyak berkembang pada kasus berkas

gambar, sedangkan pada berkas video masih sedikit [2]. Padahal, tingkat kepercayaan publik terhadap video lebih tinggi dibandingkan tingkat kepercayaan terhadap gambar. Dalam dunia hukum misalnya, bukti berupa rekaman video memiliki tingkat kepercayaan yang cukup tinggi dan sering menjadi bukti kunci untuk mengungkap kasus karena proses manipulasi video dianggap lebih sulit dibandingkan proses manipulasi gambar. Namun saat ini proses manipulasi video dipermudah dengan ketersediaan aplikasi-aplikasi video editing, baik yang open source maupun berbayar. Aplikasi-aplikasi tersebut memudahkan seseorang untuk memanipulasi video dan hasil manipulasinya dapat terlihat sangat nyata sehingga dapat dipercaya. Walaupun secara waktu, proses manipulasi video memakan waktu lebih lama dibandingkan proses manipulasi gambar. Hal ini menuntut adanya penelitian yang fokus untuk mencari solusi bagaimana mendeteksi adanya proses manipulasi terhadap sebuah video atau tidak.

Secara umum, metode deteksi manipulasi pada video terbagi ke dalam dua kategori [2]. Kategori pertama adalah tampering detection, yaitu metode deteksi yang hanya mengecek integritas dari video tanpa menunjukkan bagian mana pada video tersebut yang dimanipulasi.

Kategori kedua adalah tampering localization, yaitu metode deteksi yang menunjukkan bagian pada video yang dimanipulasi.

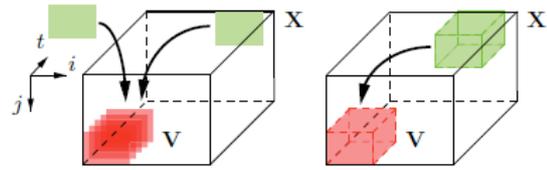
Algoritma pada kategori pertama mendeteksi proses manipulasi secara global. Contoh algoritma pada kategori ini adalah algoritma yang mendeteksi adanya kompresi ganda yang dilakukan pada sebuah berkas video [3]. Contoh lainnya adalah algoritma yang mendeteksi apakah sebuah video merupakan hasil rekam ulang dari sebuah berkas video yang lain [4]. Kemudian, pada penelitian lain dikembangkan algoritma untuk mendeteksi adanya proses interpolasi pada sebuah video [5]. Metode-metode tersebut dapat digunakan untuk mengecek validitas dari sebuah video. Namun, metode-metode tersebut tidak dapat memberikan informasi lokasi manipulasi pada video tersebut.

Algoritma pada kategori kedua dapat mendeteksi lokasi manipulasi dari video. Lokasi manipulasi berupa posisi spasial dalam bentuk koordinat pixel dan posisi temporal dalam bentuk urutan frame yang dimanipulasi. Pada penelitian [6] dikembangkan algoritma untuk mendeteksi manipulasi video tampering, yakni menyisipkan potongan sejumlah frame tertentu pada sejumlah frame yang lain. Algoritma yang dikembangkan bekerja dengan membentuk 3D block pada video dan dilakukan proses cross-correlation pada 3D block tersebut dengan seluruh bagian video secara non-overlapped. Algoritma tersebut bekerja pada level pixel. Hal ini mengakibatkan algoritma yang digunakan menjadi sangat kompleks. Karena itu pada penelitian ini diusulkan metode LBP-TOP untuk mengurangi kompleksitas algoritma deteksi video tampering tersebut. Diharapkan dengan menggunakan LBP-TOP, proses komputasi menjadi lebih sederhana dengan range nilai akurasi yang tidak terlalu jauh dengan metode sebelumnya pada [6].

2. VIDEO TAMPERING

Video tampering adalah proses untuk menyisipkan obyek tertentu ke dalam sebuah video [2]. Obyek yang disisipkan dapat berupa rangkaian frame lain dari video yang sama atau berbeda, atau rangkaian potongan frame lain dari video yang sama atau berbeda, atau sebuah gambar disisipkan ke dalam beberapa rangkaian frame.

Ilustrasi proses video tampering dapat dilihat pada gambar berikut.



Gambar 1 Ilustrasi Video tampering [6]

Dapat dilihat pada gambar terdapat dua jenis manipulasi tampering. Pada gambar yang sebelah kiri adalah proses tampering dengan menyisipkan sebuah gambar pada beberapa frame di video. Pada gambar yang sebelah kanan adalah proses tampering dengan menyisipkan potongan beberapa frame dari satu video ke beberapa frame lain yang berbeda dalam video yang sama.

Banyak metode yang telah dibangun untuk mendeteksi adanya tampering yang dilakukan terhadap video. Penelitian yang dilakukan pada [8] mendeteksi dua jenis tampering. Yang pertama adalah mendeteksi adanya spatial copy-move (duplikasi obyek yang sama pada scene yang sama) menggunakan Histogram of Gradients (HOG) matching. Yang kedua adalah mendeteksi adanya temporal copy-move (menyisipkan obyek dari sebuah frame ke frame lain) menggunakan eksploitasi struktur MPEG-2 GOP.

Penelitian yang dilakukan oleh [9] mendeteksi adanya tampering pada video menggunakan karakteristik derau. Karakteristik derau yang dimiliki oleh frame asli dengan potongan frame yang disisipkan memiliki perbedaan dan sangat sensitif terhadap proses kompresi. Pada penelitian [6] dikembangkan algoritma untuk mendeteksi manipulasi video tampering, yakni menyisipkan potongan sejumlah frame tertentu pada sejumlah frame yang lain. Algoritma yang dikembangkan bekerja dengan membentuk 3D block pada video dan dilakukan proses cross-correlation pada 3D block tersebut dengan seluruh bagian video secara non-overlapped.

3. LOCAL BINARY PATTERN-THREE ORTHOGONAL PLANES (LBP-TOP)

LBP-TOP adalah modifikasi dari metode LBP yang dapat digunakan untuk mendeskripsikan bukan hanya ciri spasial dari sebuah video, tapi juga ciri temporal. Local Binary Pattern (LBP) adalah metode untuk mendeskripsikan tekstur pada citra. LBP didefinisikan sebagai perbandingan nilai biner pixel pada pusat citra dengan nilai-nilai pixel di sekelilingnya [10]. Misalkan pada sebuah citra

Deteksi Local Tampering Pada Video Menggunakan Algoritma LBP-TOP

berukuran 3x3, nilai biner pada pusat citra dibandingkan dengan nilai-nilai di sekelilingnya. Dengan cara mengurangkan nilai pixel pada pusat citra dengan nilai-nilai pixel di sekelilingnya. Jika hasilnya lebih atau sama dengan 0 maka diberi nilai 1 dan jika hasilnya kurang dari 0 diberi nilai 0. Setelah itu, menyusun 8 nilai biner searah jarum jam atau sebaliknya dan mengubah 8 bit biner ke dalam nilai desimal untuk menggantikan nilai pixel pada pusat citra. Ilustrasi lengkap dari proses LBP adalah sebagai berikut.

g ₃	g ₂	g ₁
g ₄	g _c	g ₀
g ₅	g ₆	g ₇

Gambar 2 Ilustrasi Matriks LBP

Misalkan:

g_c = 20 g₄ = 17
 g₀ = 18 g₅ = 11
 g₁ = 23 g₆ = 121
 g₂ = 27 g₇ = 9
 g₃ = 31

Maka matriks LBP yang dihasilkan adalah sebagai berikut

1	1	1
0		0
0	1	0

Dan nilai biner dari LBP tersebut adalah 01110010.

Secara umum persamaan dari LBP dapat dituliskan sebagai berikut:

$$LBP(x_c, y_c) = \sum_{p=0}^{N-1} s(g_p - g_c) 2^p \quad (1)$$

dengan:

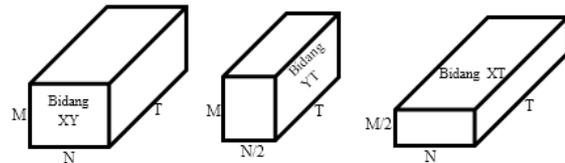
c = posisi pusat matriks LBP

N = jumlah tetangga di sekeliling pusat matriks LBP

dan s adalah fungsi dengan persamaan:

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (2)$$

Algoritma LBP-TOP akan melakukan proses ekstraksi ciri LBP pada tiga bidang[7]. Algoritma LBP-TOP dapat diilustrasikan sebagai berikut.



Gambar 3 Ilustrasi 3 Bidang LBP-TOP

Gambar di atas adalah ilustrasi dari sebuah berkas video yang terdiri dari *frame-frame* berjumlah T *frame*. Masing-masing *frame* berukuran sama, yakni MxN. LBP-TOP akan melakukan proses LBP pada bidang XY, XT dan YT. Bidang XY digunakan untuk menggambarkan ciri spasial dari video. Bidang XY diperoleh dari *frame* pertama dari video. Bidang XT dan YT digunakan untuk menggambarkan ciri temporal dari video. Bidang XT adalah bidang yang diperoleh dengan memotong balok tersebut secara melintang tepat di bagian tengah (posisi M/2). Bidang YT adalah bidang yang diperoleh dengan memotong balok tersebut secara membujur tepat di bagian tengah (posisi N/2). Dari masing-masing bidang tersebut akan muncul ciri LBP. Tiga ciri LBP ini dapat digunakan sebagai ciri dari video tersebut.

4. ANALISIS KOMPLEKSITAS CROSS CORRELATION DAN LBP-TOP

Tujuan penelitian ini adalah untuk mengusulkan metode LBP-TOP sebagai metode deteksi local tampering pada video. Metode ini diusulkan karena berdasarkan analisis kompleksitas akan lebih baik dibandingkan metode cross correlation. Analisis kompleksitas dua metode tersebut dapat dijabarkan sebagai berikut.

Misalkan dilakukan proses deteksi local tampering pada video dengan resolusi M x N pixel dengan frame sejumlah T frame. Metode cross correlation akan melakukan proses pengecekan terhadap semua pixel yang ada pada video. Maka dengan menggunakan algoritma cross correlation, kompleksitas waktu yang dihasilkan adalah = M x N x T satuan waktu. Jika diasumsikan M = N = T = n, maka T(n) adalah kompleksitas waktu dari algoritma cross correlation.

$$\begin{aligned} \text{Nilai } T(n) &= M \times N \times T \\ &= n \times n \times n \\ &= n^3 \end{aligned}$$

Maka untuk algoritma cross correlation, kompleksitas waktu yang dihasilkan adalah

$$T(n) = n^3 = O(n^3)$$

Di sisi lain, algoritma LBP-TOP akan melakukan proses ekstraksi ciri pada tiga bidang, yakni bidang XY yang berukuran M x N pixel, XT yang berukuran M x T pixel dan bidang YT yang berukuran N x T pixel. Dengan mengasumsikan $M = N = T$, maka $T(n)$ untuk algoritma LBP-TOP dapat dijabarkan sebagai berikut.

$$\begin{aligned} \text{Nilai } T(n) &= (M \times N) + (M \times T) + (N \times T) \\ &= (n \times n) + (n \times n) + (n \times n) \\ &= n^2 + n^2 + n^2 \\ &= 3n^2 \end{aligned}$$

Maka untuk algoritma LBP-TOP, kompleksitas waktu yang dihasilkan adalah

$$T(n) = 3n^2 = O(n^2)$$

Dapat terlihat bahwa dari kompleksitas waktu asimptotik kedua algoritma tersebut, algoritma LBP-TOP yang memiliki kompleksitas asimptotik $O(n^2)$ lebih efisien dibandingkan algoritma cross correlation yang memiliki kompleksitas asimptotik $O(n^3)$. Hasil analisis ini akan diperkuat pada tahap eksperimen.

5. RANCANGAN EKSPERIMEN

Tujuan eksperimen ini adalah untuk membuktikan hasil analisis perbandingan kompleksitas metode cross correlation dan metode LBP-TOP untuk mendeteksi local tampering pada video. Eksperimen dilakukan pada dua video yang mengalami local tampering. Adapun tahap dari eksperimen ini adalah sebagai berikut.

- Menandai lokasi terjadinya tampering pada kelima video.
- Menjalankan sistem deteksi local tampering menggunakan algoritma cross correlation dan LBP-TOP. Kedua algoritma tersebut dijalankan dengan terlebih dahulu melakukan pemotongan video menjadi sub-blok, lalu dilakukan pencocokan antar sub-blok menggunakan kedua algoritma tersebut.

- Mengamati hasil deteksi dari kedua algoritma tersebut dan mencatat waktu eksekusi kedua algoritma tersebut.

- Menganalisis perbandingan waktu eksekusi kedua algoritma.

Adapun parameter yang diujikan pada eksperimen tersebut adalah sebagai berikut.

- Ukuran panjang dan lebar sub-blok: 40 dan 20 pixel
- Jumlah frame sub-blok: 60 dan 30 frame
- Overlap panjang & lebar sub-blok: 0% (tanpa overlap), 50% (overlap separuh)
- Overlap jumlah frame sub-blok: 0% (tanpa overlap), 50% (overlap separuh)

6. HASIL EKSPERIMEN DAN KESIMPULAN

Berikut adalah hasil dari eksperimen yang dilakukan pada kedua video. Masing-masing table menunjukkan hasil pada masing-masing video. Kolom pertama menunjukkan ukuran sub-blok. Kolom kedua menunjukkan jumlah frame yang digunakan sebagai sub-blok. Kolom ketiga menunjukkan berapa ukuran overlap sub-blok. Kolom keempat menunjukkan berapa overlap jumlah frame. Kolom kelima adalah waktu eksekusi dari algoritma cross correlation. Kolom keenam adalah waktu eksekusi dari algoritma LBP-TOP.

Tabel 1 Hasil Eksperimen pada Video 1

Ukuran sub-blok	Jumlah Frame	Overlap Ukuran sub-blok	Overlap Jumlah Frame	Waktu eksekusi Cross Correlation	Waktu eksekusi LBP-TOP
40	60	0%	0%	974.38	188.34
40	60	0%	50%	2015.62	524.83
40	60	50%	0%	16868.29	1688.95
40	60	50%	50%	35053.58	6129.12
40	30	0%	0%	4138.44	776.02
40	30	0	50%	7687.69	2468.78
40	30	50%	0%	47758.15	7605.62
40	30	50%	50%	75201.53	22440.78
20	60	0%	0%	16315.23	1605.29
20	60	0%	50%	41285.66	7547.99
20	60	50%	0%	255663	23582.80

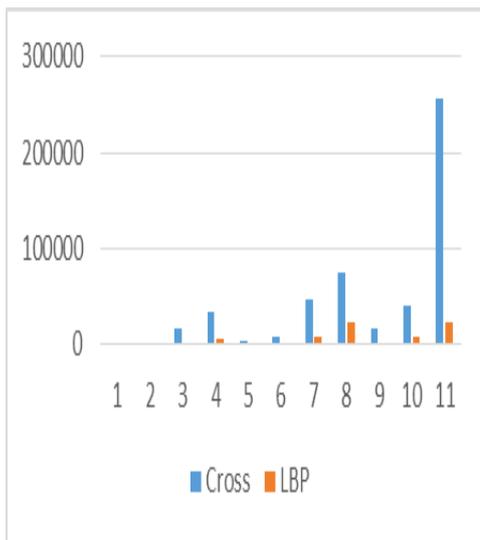
Deteksi Local Tampering Pada Video Menggunakan Algoritma LBP-TOP

Tabel 2 Hasil Eksperimen pada Video 2

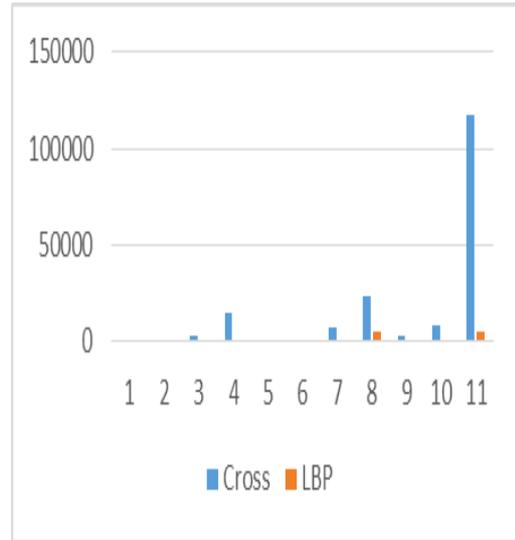
Ukuran sub-blok	Jumlah Frame	Overlap Ukuran sub-blok	Overlap Jumlah Frame	Waktu eksekusi Cross Correlation	Waktu eksekusi LBP-TOP
40	60	0%	0%	241.84	51.19
40	60	0%	50%	491.76	139.87
40	60	50%	0%	2849.60	339.14
40	60	50%	50%	14449.24	1157.27
40	30	0%	0%	586.91	130.32
40	30	0	50%	1205.06	448.39
40	30	50%	0%	6968.90	1244.94
40	30	50%	50%	23610.59	4836.46
20	60	0%	0%	3127.51	379.85
20	60	0%	50%	8519.75	1380.90
20	60	50%	0%	117148	4836.38

Dari kedua tabel di atas, terlihat bahwa waktu eksekusi dari algoritma cross correlation selalu lebih tinggi dibandingkan algoritma LBP-TOP. Artinya, algoritma LBP-TOP bekerja lebih efisien dari sisi waktu jika dibandingkan dengan algoritma Cross Correlation.

Berikut adalah grafik performansi waktu dari kedua algoritma berdasarkan data pada tabel di atas. Sumbu horizontal menyatakan urutan konfigurasi parameter sesuai data pada tabel. Sumbu vertikal menyatakan waktu eksekusi dari kedua algoritma. Grafik biru mewakili algoritma cross correlation dan grafik merah mewakili algoritma LBP-TOP.



Gambar 4 Grafik Performansi Waktu pada Video 1



Gambar 4 Grafik Performansi Waktu pada Video 2

Dari data tersebut di atas, dapat disimpulkan bahwa hasil analisis kompleksitas waktu yang dijabarkan terbukti benar. Langkah berikutnya dari penelitian ini yang akan dilakukan adalah menganalisis lebih lanjut tentang deteksi local tampering video menggunakan algoritma LBP-TOP dan meneliti bagaimana hasil akurasi dari metode tersebut.

DAFTAR PUSTAKA

- [1] R. Poisel and S. Tjoa, "Forensics investigations of multimedia data: A review of the state-of-the-art," *IT Security Incident Management and IT Forensics (IMF)*, 2011.
- [2] S. Milani et al., "An overview on video forensics," *APSIPA Transactions on Signal and Information Processing*, vol. 1, p. E2, 2012.
- [3] P. Bestagini, M. Tagliasacchi, S. Tubaro S. Milani, "Multiple compression detection for video sequences," in *2012 IEEE 14th International Workshop on Multimedia Signal Processing (MMSP)*, 2012.
- [4] M. Visentini-Scarzanella, M. Tagliasacchi, P. Dragotti, S. Tubaro P. Bestagini, "Video recapture detection based on ghosting artifact analysis," in *2013 IEEE International Conference on Image Processing (ICIP)*, 2013.
- [5] S. Battaglia, S. Milani, M. Tagliasacchi, S. Tubaro P. Bestagini, "Detection of temporal interpolation in video sequences,"

- in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.
- [6] Paolo Bestagini, Simone Milani, and Marco Tagliasacchi, "Local tampering detection in video sequences," in *15th IEEE International Workshop on Multimedia Signal Processing*, Pula (Sardinia), 2013.
- [7] M. Pietikainen, T. Maenpaa T. Ojala, "Multiresolution Gray Scale and Rotation Invariant Texture Analysis with Local Binary Patterns," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987, 2002.
- [8] A. Subramanyam and S. Emmanuel, "Video forgery detection using HOG features and compression properties," in *2012 IEEE 14th International Workshop on Multimedia Signal Processing (MMSP)*, 2012.
- [9] T. Okabe, Y. Sato M. Kobayashi, "Detecting forgery from staticscene video based on inconsistency in noise level functions," *IEEE Transactions on Information Forensics and Security*, vol. 5, pp. 883–892, 2010.
- [10] G. Zhao M. Pietikainen, "Local Binary Pattern Descriptors for Dynamic Texture Recognition," in *International Conference of Pattern Recognition*, 2006, pp. 211-214.

TEKNIK DATA MINING UNTUK MEMPREDIKSI MASA STUDI MAHASISWA MENGUNAKAN ALGORITMA K-NEAREST NEIGHBORHOOD

Selvia Lorena Br Ginting¹⁾, Wendi Zarman²⁾, Astrid Darmawan³⁾

^{1,2,3)} Jurusan Teknik Komputer, Fakultas Teknik dan Ilmu Komputer

Universitas Komputer Indonesia

Jl. Dipatiukur No.112-116 Bandung 40132

e-mail: selvialorena@yahoo.com

ABSTRAK

Data mining adalah sebuah teknik yang memanfaatkan data dalam jumlah yang besar untuk memperoleh informasi berharga yang sebelumnya tidak diketahui dan dapat dimanfaatkan untuk pengambilan keputusan penting. Data mining juga memanfaatkan pengalaman atau bahkan kesalahan di masa lalu untuk meningkatkan kualitas dari model maupun hasil analisisnya, salah satunya dengan kemampuan pembelajaran yang dimiliki teknik data mining yaitu klasifikasi. Kegiatan pengklasifikasian yang dilakukan oleh manusia masih memiliki keterbatasan, terutama pada kemampuan manusia dalam menampung jumlah data yang ingin diklasifikasikan. Selain itu bisa juga terjadi kesalahan dalam pengklasifikasian yang dilakukan. Salah satu cara mengatasi masalah ini adalah dengan menggunakan Data Mining (DM) dengan teknik klasifikasi. Klasifikasi merupakan tugas pembelajaran yang memetakan sebuah objek baru ke dalam salah satu label class atau kategori pada objek lama yang telah didefinisikan sebelumnya. Klasifikasi ini menggunakan salah satu metode algoritma data mining yaitu k-Nearest Neighborhood (KNN). Algoritma KNN bekerja berdasarkan jarak terdekat dari objek baru ke objek lama dengan menentukan nilai k. Nilai k merupakan parameter untuk menentukan jarak terdekat antara objek baru terhadap objek lama. Dengan menggunakan teknik data mining tersebut maka di perguruan tinggi dapat memanfaatkan data akademik mahasiswa yaitu indeks prestasi (IP) untuk memprediksi masa studi mahasiswa berdasarkan kategori kelulusan yaitu tepat waktu (4-5 Tahun) dan tidak tepat waktu (5 tahun lebih). Dalam aplikasi data mining ini terdiri dari data testing (data yang akan diuji) dan data training (data yang telah diketahui label class atau kategorinya) dengan masukan NIM dan nilai k. Nilai k yang terbaik, tergantung pada jumlah data yang digunakan. Jika nilai k tinggi, maka hasil tingkat keberhasilannya belum tentu baik dan begitu sebaliknya. Sehingga diharapkan hasil akhir dari aplikasi data mining ini dapat menghasilkan prediksi masa studi mahasiswa.

Kata Kunci: Data Mining, Klasifikasi, Algoritma k-Nearest Neighborhood, Prediksi Masa Studi Mahasiswa

1. PENDAHULUAN

Dalam dunia pendidikan terutama pendidikan tinggi, data yang berlimpah dan berkesinambungan mengenai mahasiswa yang dibina dan alumni terus dihasilkan. Pertumbuhan yang pesat dari penambahan data akademik ini telah menciptakan kondisi dimana suatu perguruan tinggi memiliki tumpukan data yang banyak. Namun pada saat ini, tumpukan data tersebut banyak yang belum dimanfaatkan secara maksimal bahkan tidak terpakai. Padahal tumpukan data tersebut dapat menjadi sebuah informasi yang bermanfaat dengan menggunakan suatu teknik yaitu teknik *data mining*. *Data mining* adalah serangkaian proses untuk menggali nilai tambah berupa informasi yang selama ini tidak diketahui secara manual dari suatu *database*. Penggunaan teknik *data mining* pada

perguruan tinggi dapat berguna mengolah dan menyebarkan informasi untuk menunjang kegiatan operasional sehari-hari sekaligus menunjang kegiatan pengambilan keputusan strategis. Data yang berlimpah tersebut membuka peluang diterapkannya *data mining* untuk pengelolaan pendidikan yang lebih baik dan *data mining* dalam pelaksanaan pembelajaran berbantuan komputer yang lebih efektif dalam suatu perguruan tinggi.

Penelitian ini memanfaatkan data akademik yang sebelumnya hanya menjadi beban *database* yang dimiliki oleh jurusan Teknik Komputer UNIKOM, yaitu data IP (Indeks Prestasi) mahasiswa dari semester satu sampai semester enam khususnya Program Sarjana (S1). Data ini akan dimanfaatkan sebagai sumber informasi strategis bagi jurusan untuk memprediksi masa studi mahasiswa dengan menerapkan salah satu teknik dari *data*

mining yaitu klasifikasi dengan algoritma *k-Nearest-Neighborhood*. Hal ini dilakukan dengan harapan dapat menemukan informasi tingkat kelulusan dan persentase kelulusan mahasiswa sehingga dapat digunakan oleh pihak jurusan untuk mencari solusi atau kebijakan dalam proses evaluasi pembelajaran di Jurusan Teknik Komputer. Tujuan yang ingin dicapai dalam pembangunan aplikasi *data mining* ini dapat memprediksi masa studi mahasiswa sehingga dapat mengetahui tingkat kelulusan dan persentase kelulusan mahasiswa di jurusan Teknik Komputer.

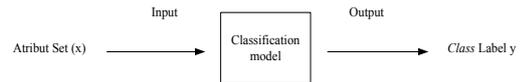
2. LANDASAN TEORI

Data Mining

Data mining merupakan proses menemukan pengetahuan yang menarik dari data yang berjumlah besar yang disimpan di dalam *database*, gudang data atau repositori informasi. *Data mining* juga disebut sebagai serangkaian proses untuk menggali nilai tambah berupa pengetahuan yang selama ini tidak diketahui secara manual dari suatu kumpulan data. Secara umum, *data mining* dapat melakukan dua hal yaitu memberikan kesempatan untuk menemukan informasi menarik yang tidak terduga dan juga bisa menangani data berskala besar. Dalam menemukan informasi yang menarik ini, ciri khas *data mining* adalah kemampuan pencarian secara hampir otomatis, karena dalam banyak teknik *data mining* ada beberapa parameter yang masih harus ditentukan secara manual atau semi manual. *Data mining* juga dapat memanfaatkan pengalaman atau bahkan kesalahan di masa lalu untuk meningkatkan kualitas dari model maupun hasil analisisnya, salah satunya dengan kemampuan pembelajaran yang dimiliki beberapa teknik *data mining* seperti klasifikasi.

Klasifikasi

Klasifikasi adalah tugas pembelajaran sebuah fungsi target f yang memetakan setiap himpunan atribut x ke salah satu label *class* y yang telah didefinisikan sebelumnya. Klasifikasi dapat juga diartikan suatu proses untuk menemukan suatu model atau fungsi yang menggambarkan dan membedakan kelas data atau konsep dengan tujuan dapat menggunakan model untuk memprediksi kelas objek yang label *class*-nya tidak diketahui.



Gambar 1. Model Klasifikasi

Data input untuk klasifikasi adalah isi dari *record*. Setiap *record* dikenal sebagai *instance* atau contoh, yang ditentukan oleh sebuah *tuple* (x, y) , dimana x adalah himpunan atribut dan y adalah atribut tertentu, yang dinyatakan sebagai label *class* (juga dikenal sebagai kategori atau atribut target).

Pendekatan umum yang digunakan dalam masalah klasifikasi adalah pertama, *data testing* berisi *record* yang mempunyai label *class* yang telah diketahui. *Data training* digunakan untuk membangun model klasifikasi yang kemudian diaplikasikan ke *data testing* yang berisi *record-record* dengan label *class* yang tidak diketahui.

Algoritma Nearest Neighborhood

Algoritma *Nearest Neighborhood* adalah pendekatan untuk mencari kasus dengan menghitung kedekatan antara kasus baru (*data testing*) dengan kasus lama (*data training*), yaitu berdasarkan pada pencocokan bobot dari sejumlah fitur yang ada. Jenis algoritma *Nearest Neighborhood* ada 2, yaitu:

1. 1-NN, yaitu pengklasifikasikan dilakukan terhadap 1 *labeled data* terdekat.
2. k -NN, yaitu pengklasifikasikan dilakukan terhadap k *labeled data* terdekat dengan $k > 1$.

Di dalam penelitian ini akan digunakan Algoritma *k-Nearest Neighborhood*.

Algoritma k-Nearest Neighborhood (k-NN)

Algoritma *k-Nearest Neighborhood (k-NN)* adalah suatu metode yang menggunakan algoritma *supervised* dimana hasil dari *query instance* yang baru diklasifikasi berdasarkan mayoritas dari label *class* pada k -NN. Tujuan dari algoritma k -NN adalah mengklasifikasi objek baru berdasarkan atribut dan *data training*.

Algoritma k -NN bekerja berdasarkan jarak terpendek dari *query instance* ke *data training* untuk menentukan k -NN-nya. Salah satu cara untuk menghitung jarak dekat atau jauhnya tetangga menggunakan metode *euclidian distance*.

Teknik Data Mining Untuk Memprediksi Masa Studi Mahasiswa Menggunakan Algoritma K-Nearest Neighborhood

Ecludian Distance sering digunakan untuk menghitung jarak. *Euclidian Distance* berfungsi menguji ukuran yang bisa digunakan sebagai interpretasi kedekatan jarak antara dua obyek, di bawah ini merupakan rumus *Ecludian Distance*:

$$\left(\sum_{k=1}^m (x_{ik} - x_{jk})^2 \right)^{1/2}$$

Dimana,

X_{ik} = nilai X pada *data training*

X_{jk} = nilai X pada *data testing*

m = batas jumlah banyaknya data

Jika hasil nilai dari rumus di atas besar maka akan semakin jauh tingkat keserupaan antara kedua objek dan sebaliknya jika hasil nilainya semakin kecil maka akan semakin dekat tingkat keserupaan antar objek tersebut. Objek yang dimaksud adalah *data training* dan *data testing*.

Dalam algoritma ini, nilai *k* yang terbaik itu tergantung pada jumlah data. Ukuran nilai *k* yang besar belum tentu menjadi nilai *k* yang terbaik begitupun juga sebaliknya.

Langkah-langkah untuk menghitung algoritma *k*-NN:

1. Menentukan nilai *k*.
2. Menghitung kuadrat jarak *euclid (query instance)* masing-masing objek terhadap *data training* yang diberikan.
3. Kemudian mengurutkan objek-objek tersebut ke dalam kelompok yang mempunyai jarak *euclid* terkecil.
4. Mengumpulkan label *class Y* (klasifikasi *Nearest Neighborhood*).
5. Dengan menggunakan kategori *Nearest Neighborhood* yang paling mayoritas maka dapat diprediksikan nilai *query instance* yang telah dihitung.

3. ANALISIS DAN PERANCANGAN PERANGKAT LUNAK

Analisis Data

UNIKOM merupakan salah satu perguruan tinggi swasta di Bandung. UNIKOM memiliki beberapa Jurusan salah satunya adalah Jurusan Teknik Komputer. Jurusan ini termasuk kategori yang sangat sulit untuk lulus tepat waktu. Setiap tahun, Jurusan Teknik Komputer hanya menghasilkan beberapa mahasiswa yang lulus 4 atau 5 tahun.

Karena jumlah kelulusan tiap tahunnya hanya sedikit, maka peneliti memanfaatkan data nilai IP mahasiswa di Jurusan Teknik Komputer untuk menemukan informasi atau pengetahuan baru yang berguna dalam mengambil sebuah keputusan dan membantu dalam evaluasi sistem pembelajaran di Jurusan Teknik Komputer. Informasi yang dibutuhkan adalah memprediksi masa studi mahasiswa dengan atribut IP dari semester satu sampai semester enam.

Data akademik mahasiswa yang diambil adalah data mahasiswa angkatan 2001-2006. Hal ini didasarkan pada kebutuhan data yang akan dihubungkan dengan *data testing*, dengan asumsi bahwa mahasiswa angkatan 2001-2006 akan lulus dari rentang waktu tahun 2005-2010. Sedangkan data kelulusan dalam *data training* rentang waktunya dari tahun 2004-2011.

Aplikasi data mining yang dibuat terdiri dari dua data, yaitu:

1. *Data Testing*
 - a. NIM
 - b. Indeks Prestasi (IP) mahasiswa dari semester satu sampai enam.
2. *Data Training*
 - a. NIM
 - b. Indeks Prestasi (IP) mahasiswa dari semester satu sampai enam.
 - c. Keterangan (Kategori Kelulusan)

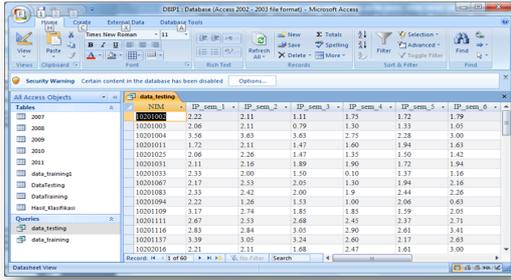
Data training memiliki kategori sebagai berikut:

Tabel 1. Kategori Kelulusan Berdasarkan Lama Studi

Lulus	Kategori
Lulus <= 5 Tahun	Ya
Lulus > 5 Tahun	Tidak

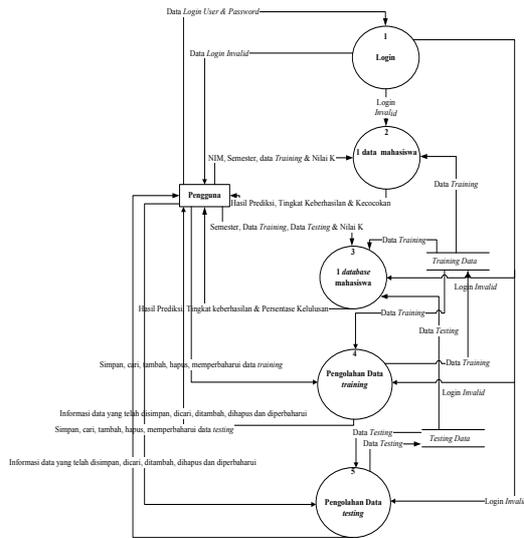
Tahun	NIM	IP_sem_1	IP_sem_2	IP_sem_3	IP_sem_4	IP_sem_5	IP_sem_6	Keterangan
2007	10200001	3,00	3,11	1,89	1,70	2,22	2,21	Tidak
2008	10200004	2,72	2,32	1,89	1,60	2,33	2,11	Tidak
2009	10200005	3,28	3,37	3,32	1,60	1,83	2,00	Ya
2010	10200003	4,00	3,79	4,00	3,35	3,17	3,42	Ya
2011	10200024	3,61	3,16	2,89	2,36	2,67	2,38	Ya
	10200028	3,66	2,84	1,68	2,15	2,11	1,68	Ya
	10200039	3,72	3,38	3,84	3,35	2,94	3,11	Tidak
	10200040	2,17	2,16	1,29	1,65	1,44	1,68	Tidak
	10200041	1,50	1,89	0,93	1,65	2,00	1,84	Tidak
	10200044	3,50	3,42	3,27	2,45	2,83	2,84	Ya
	10200048	2,84	2,95	2,21	2,00	2,36	2,35	Tidak
	10200049	3,84	3,89	3,53	3,55	3,39	3,68	Ya
	10200061	3,44	3,38	2,77	2,20	2,56	2,95	Ya
	10200074	2,83	2,63	2,58	2,35	2,67	2,82	Tidak
	10200079	3,11	3,31	2,89	3,25	3,22	2,92	Ya

Gambar 2. Cuplikan *Data Training* (61 data)

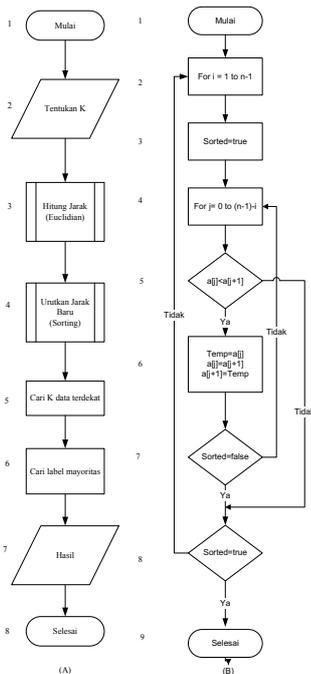


Gambar 3. Cuplikan Data Testing (30 data)

Data Flow Diagram Sistem



Flowchart Sistem



Gambar 4. (A) Algoritma KNN & (B) Algoritma Sorting

Tabel 2. Penjelasan Flowchart Algoritma k-NN

Urutan	Keterangan
1	Memulai awal program.
2	Menentukan nilai k .
3	Pemanggilan procedure untuk menghitung jarak baru menggunakan rumus <i>euclidean</i> .
4	Pemanggilan procedure untuk mengurutkan jarak baru menggunakan algoritma <i>sorting</i> , yaitu <i>insertion</i> .
5	Mencari jarak terdekat sesuai nilai k .
6	Mencari mayoritas label <i>class</i> pada jarak terdekat sesuai nilai k .
7	Menghasilkan prediksi.
8	Program selesai.

Tabel 3. Penjelasan Flowchart Algoritma Sorting

Urutan	Keterangan
1	Memulai <i>sorting</i> .
2	Pengulangan inisialisasi i dari 1 hingga $(n-1)$.
3	<i>Sorting</i> inisialisasi i benar.
4	Pengulangan inisialisasi j dari 0 hingga $((n-1)-i)$.
5	Apakah isi $a[j] < a[j+1]$? Jika Ya, maka melakukan pertukaran data dan jika Tidak, maka melanjutkan proses di urutan 8.
6	Melakukan pertukaran isi data yang terkecil hingga terbesar.
7	Jika <i>sorting</i> -nya salah, maka kembali ke urutan 4.
8	Apakah <i>sorting</i> -nya sudah benar? Jika Tidak, maka mengulang proses urutan 2 dan jika Ya, maka <i>sorting</i> selesai
9	Proses <i>sorting</i> selesai.

4. PENGUJIAN SISTEM

Pengujian berguna untuk mengukur kehandalan dari sistem atau alat yang dibangun, sehingga hasil yang diharapkan dapat sesuai dengan yang dibutuhkan.

Pengujian dilakukan terhadap aplikasi *data mining* yang dibangun untuk melihat apakah aplikasi ini berhasil atau tidak dalam memprediksi masa studi mahasiswa. Pengujian terdiri dari 2 proses yaitu:

Teknik Data Mining Untuk Memprediksi Masa Studi Mahasiswa Menggunakan Algoritma K-Nearest Neighborhood

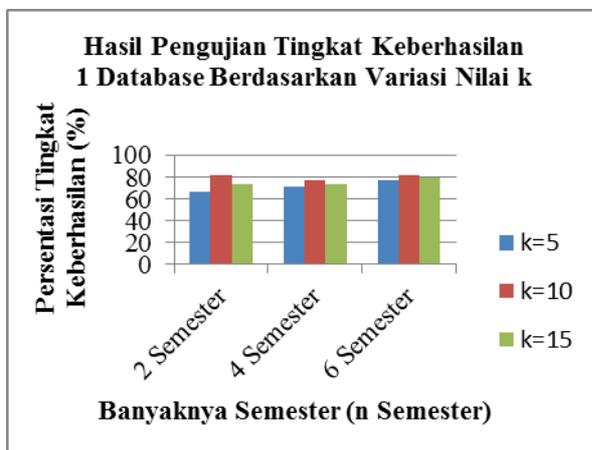
1. Pengujian 1 *database* mahasiswa (keseluruhan) dengan menggunakan *data training* yang berjumlah 30 data.
2. Pengujian 1 *database* mahasiswa (keseluruhan) dengan menggunakan *data training* yang berjumlah 61 data.

Masing-masing proses pengujian tersebut menggunakan Indeks Prestasi (IP) dua semester (semester 1 dan 2), empat semester (semester 1-4) dan enam semester (semester 1-6) dengan menggunakan nilai k yang berbeda. Pengujian ini dilakukan dengan menggunakan *data training* yang berjumlah sebanyak 61 data dan 30 data serta *data testing* yang berjumlah 60 data. Hasil prediksi 1 *database* akan dibandingkan dengan data asli dan dicari kecocokannya secara otomatis oleh program.

Pengujian ini juga berguna untuk mengetahui apakah nilai k yang digunakan adalah nilai k yang terbaik dengan hasil tingkat keberhasilannya tinggi atau tidak untuk memprediksi kelulusan mahasiswa pada sistem aplikasi *data mining* ini.

Untuk mengetahui tingkat keberhasilan pada sistem ini digunakan rumus:

$$\frac{\sum \text{hasil pengujian bernilai benar}}{\sum \text{banyaknya data sampel}} \times 100\%$$



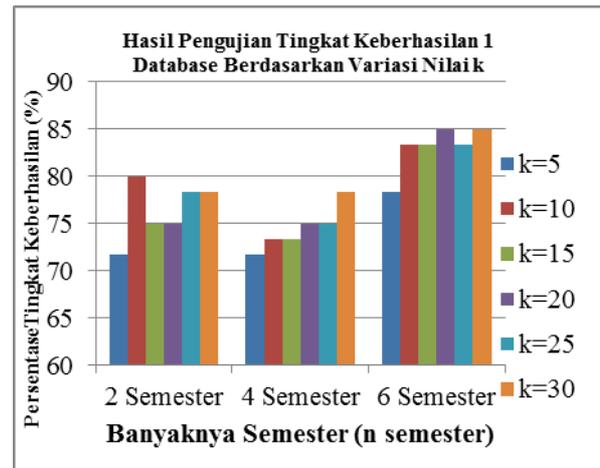
Gambar 5. Grafik Pengujian 1 *database* dengan *Data Training* = 30 data

Dari hasil gambar 5 dapat disimpulkan sebagai berikut:

1. Untuk dua semester yaitu nilai k yang terbaik untuk digunakan memprediksi studi mahasiswa adalah nilai k=10 dengan tingkat keberhasilan 81.66%.
2. Untuk empat semester yaitu nilai k yang terbaik untuk digunakan memprediksi

studi mahasiswa adalah nilai k=10 dengan tingkat keberhasilan 76.66%.

3. Untuk enam semester yaitu nilai k=10 merupakan nilai k yang terbaik untuk digunakan memprediksi masa studi mahasiswa dengan tingkat keberhasilan 81.66%.
4. Masing-masing nilai k terbaik di atas hanya berlaku untuk *data training* yang berjumlah 30 data dan *data testing*-nya 60 data.



Gambar 6. Grafik Pengujian 1 *database* dengan *Data Training* = 61 data

Dari hasil Gambar 6 dapat disimpulkan sebagai berikut:

1. Untuk dua semester yaitu nilai k yang terbaik untuk memprediksi masa studi mahasiswa adalah nilai k=10 dengan tingkat keberhasilan 80%.
2. Untuk empat semester, nilai k yang terbaik untuk memprediksi masa studi mahasiswa adalah nilai k=30 dengan tingkat keberhasilan 78.33%.
3. Untuk enam semester, nilai k yang terbaik untuk memprediksi masa studi mahasiswa adalah nilai k=20 dan k=30 dengan tingkat keberhasilan 85%.
4. Masing-masing nilai k terbaik di atas hanya berlaku untuk *data training* yang berjumlah 61 data dan *data testing*-nya 60 data.

Pada percobaan satu dan dua di atas dapat dilihat bahwa dengan mengubah nilai k akan menghasilkan data yang bervariasi. Jadi, ukuran nilai k yang besar untuk digunakan memprediksi masa studi mahasiswa belum tentu menjadi nilai k yang terbaik dengan tingkat keberhasilan yang tinggi begitupun

juga sebaliknya. Nilai k yang terbaik dipengaruhi oleh jumlah data yang digunakan.

5. KESIMPULAN

Berdasarkan pemaparan diatas, dapat diambil beberapa kesimpulan, yaitu:

1. *Data training* dengan jumlah 30 data digunakan untuk menguji *data testing* berjumlah 30 data, didapatkan nilai k yang terbaik untuk memprediksi masa studi mahasiswa yaitu sebagai berikut:
 - 1) Untuk dua semester yaitu nilai k yang terbaik untuk digunakan memprediksi studi mahasiswa adalah nilai $k=10$ dengan tingkat keberhasilan 81.66%.
 - 2) Untuk empat semester yaitu nilai k yang terbaik untuk digunakan memprediksi studi mahasiswa adalah nilai $k=10$ dengan tingkat keberhasilan 76.66%. Untuk enam semester yaitu nilai $k=10$ merupakan nilai k yang terbaik untuk digunakan memprediksi masa studi mahasiswa dengan tingkat keberhasilan 81.66%.
2. *Data training* dengan jumlah 60 data digunakan untuk menguji *data testing* berjumlah 61 data, diperoleh nilai k yang terbaik untuk memprediksi masa studi mahasiswa yaitu sebagai berikut:
 - 1) Untuk dua semester yaitu nilai k yang terbaik untuk memprediksi masa studi mahasiswa adalah nilai $k=10$ dengan tingkat keberhasilan 80%.
 - 2) Untuk empat semester, nilai k yang terbaik untuk memprediksi masa studi mahasiswa adalah nilai $k=30$ dengan tingkat keberhasilan 78.33%.
 - 3) Untuk enam semester, nilai k yang terbaik untuk memprediksi masa studi mahasiswa adalah nilai $k=20$ dan $k=30$ dengan tingkat keberhasilan 85%.

6. DAFTAR PUSTAKA

- [1] Han, J., & Kamber, M. (2006). *Data Mining Concepts and Techniques*. Second Edition. San Fransisco: Morgan Kauffman.
- [2] Huda, Masykur, N. (2010). *Aplikasi Data Mining untuk Menampilkan Informasi Tingkat Kelulusan Mahasiswa*. Semarang: Universitas Diponegoro.
- [3] Keller, J.M., Gray, M.R., & Givens, J.A (1985). *A Fuzzy k-Nearest Neighbor*

Algorithm. Systems, Man and Cybermatics, IEEE Transactions. (4). 580-585.

- [4] Kusriani & Luthfi, E., T. (2009). *Algoritma Data Mining*. Yogyakarta: Andi.
- [5] Rismawan, Tedy, et al. (2008). *Sistem Pendukung Keputusan Berbasis Pocket PC Sebagai Penentu Status Gizi Menggunakan Metode KNN (K-Nearest Neighbor)*. Jurnal Teknoin.
- [6] Seidl, Thomas, and Kriegl, H. P. (1998). *Optimal Multi-Step k-Nearest Neighbor Search*. ACM SIGMOD Record. Vol 27. No. 2. ACM.

Panduan Penulisan

- Naskah ditulis dalam Bahasa Indonesia. Diketik dalam 2 kolom (ukuran kertas A4) dengan huruf Times New Roman ukuran 11, rata kanan-kiri 2.5 cm.
- Naskah diawali dengan judul, nama penulis (tanpa gelar), instansi dan alamat email untuk korespondensi. Isi naskah setidaknya berisi/menerangkan tentang pendahuluan, metoda, hasil diskusi, kesimpulan, daftar pustaka. Ucapan terima kasih bila diperlukan dapat dituliskan setelah bagian kesimpulan.
- Naskah dilengkapi dengan abstrak (100-250 kata) diikuti dengan kata kunci (4-8 kata) dalam bahasa Indonesia. Abstrak hendaknya menjelaskan tentang apa yang diteliti/ dikembangkan/ dipikirkan, metoda dan hasil sehingga isi naskah dapat tergambar tanpa membaca seluruh tulisan.
- Judul bagian naskah serta sub judul untuk bagian berbeda (pendahuluan, metoda, hasil, diskusi, kesimpulan) diberikan nomor arab secara bertingkat (1, 1.1., dst)
- Satuan ukuran yang digunakan serta singkatannya mengikuti aturan International System of Units (SI).
- Persamaan dinomori berurutan dengan nomor arab dalam kurung disebelah kanan persamaan (rata kanan). Penggunaan simbol dalam persamaan diberi keterangan dalam bagian naskah dibawahnya.
- Penyisipan gambar dan tabel disertai dengan judul singkat dibawahnya (untuk gambar) dan diatasnya (untuk tabel) serta diberi nomor berurutan. Seluruh gambar dan tabel yang digunakan harus diulas pada bagian tulisan.
- Dari daftar referensi yang dikutip dituliskan pada bagian akhir naskah dengan memberikan nomor urut sesuai dengan urutan pengutipan pada naskah. Bagian naskah yang mengacu pada satu atau beberapa literatur lain hendaknya mencantumkan nomor urut referensi pada daftar pustaka.
- Contoh penulisan referensi adalah sebagaimana berikut:
 - [1] Orfarindis, Sopocles J. *Optimum Signal Processing, An Inroduction*, 2nd Edition, McGraw-Hill Book Company, 1988.
 - [2] LIN, David W. , *On Digital Implementation of the Fast Kalman Algorithms*, IEEE Transaction on Acoustics, Speech and Signal Processing, vol. ASSP-32, hal 998-1005, No.5, Oktober 1984.
 - [3] All data sheet, *Hitachi HM55B Compas Module (#29213)*. Diakses 05 Juli 2011, dari <http://pdf1.alldatasheet.com/datasheet-pdf/view/228104/ETC2/HM55B/+Q2J227VwSw9bIPvzH+/datasheet.pdf>



Jurusan Teknik Komputer Unikom
Jl. Dipati Ukur No. 112
Bandung 40132
Tel / fax : (022) 2504119 / (022) 2533754

ISSN 2252-9039

