



Melek IT

Program Studi
Teknik Informatika
Universitas Wijaya Kusuma Surabaya

Jurnal Teknologi Informasi Dan Komunikasi

Volume 3 No 2 Juli 2014

ISSN 2252-9128



APLIKASI KAMUS BAHASA INDONESIA-JAWA MENGGUNAKAN SEQUENTIAL SEARCHING PADA PLATFORM ANDROID. Erfan Rachmad Santosa, Nia Saurina.

PERANCANGAN MODEL TATA KELOLA TEKNOLOGI INFORMASI PADA FAKULTAS TEKNIK UNIVERSITAS WIJAYA KUSUMA SURABAYA BERBASIS COBIT FRAMEWORK DOMAIN PLANNING AND ORGANIZING. Emmy Wahyuningtyas.

APLIKASI PENGHITUNGAN PAJAK PENGHASILAN BERBASIS ANDROID DENGAN STUDI KASUS RSUD BHAKTI DHARMA HUSADA SURABAYA. Fajar Wahyudi, Anang Kukuh.A

MEMBANGUN SERVER VOIP TRIXBOX DENGAN CLIENT ANDROID.
Muhammad Faizal , Tjatur Widiartin.

MENINGKATKAN KINERJA METODE DETEKSI TEPI FUZZY MENGGUNAKAN HIERARCHICAL CLUSTER ANALYSIS. Saiful Nur Budiman, Yanuar Risah Prayogi, Stendy B. Sakur.

VIDEO COMPOSITING MENGGUNAKAN POISSON BLENDING
Saiful Yahya, Mochamad Hariadi, Ahmad Zaini, Eko Mulyanto Yuniarto.

PERANCANGAN PERMAINAN BERBASIS ANDROID SEBAGAI MEDIA BELAJAR UNTUK ANAK USIA DINI UMUR 2-4 TAHUN MENGGUNAKAN AUGMENTED REALITY.
Anang Kukuh.A.

PERGERAKAN KERUMUNAN MENGGUNAKAN RVO (RECIPROCAL VELOCITY OBSTACLE) WITH EXTENDED ORCA (OPTIMAL RECIPROCAL COLLISION AVOIDANCE) UNTUK NPC (NON PLAYER CHARACTER). Fariz Djunaidi, Mochamad Hariadi, Supeno mardi.

SEGMENTASI SEL DARAH PUTIH BERDASARKAN WARNA MENGGUNAKAN K-MEANS DAN OPERASI MORFOLOGI. Shofiya Syidada, Nanik Suciati, Chastine Fatichah

SURVEY PADA METODE COLOR CONSTANCY UNTUK PERBAIKAN WARNA CITRA DIPSTICK URINE.
Ali S. Kholimi, dan R.V. Hari Ginardi.

ISSN 2252-9128

Melek IT

Jurnal Teknologi Informasi Dan Komunikasi

Jurnal **MelekIT** dipublikasikan oleh Program Studi Teknik Informatika, Fakultas Teknik, Universitas Wijaya Kusuma Surabaya. Jurnal diterbitkan dua kali dalam satu tahun pada bulan Januari dan bulan Juli, berisi tulisan yang diangkat dari hasil penelitian di bidang komunikasi dan informatika.

Puji syukur senantiasa kita panjatkan ke hadirat Tuhan Yang Maha Esa terutama dengan telah terbitnya Jurnal Teknologi Informasi dan Komunikasi yang diberi nama **MelekIT**. Jurnal penelitian ini memang telah lama ditunggu kehadirannya mengiringi dengan adanya Program Studi Teknik Informatika di lingkungan Fakultas Teknik Universitas Wijaya Kusuma Surabaya, sehingga sangat perlu sebagai sarana untuk mensosialisasikan hasil-hasil penelitian dan pengembangan terutama di bidang komunikasi dan informatika.

Kami berharap semoga jurnal penelitian ini dapat menjadi sarana komunikasi antara komunitas peneliti di Indonesia serta antara peneliti dengan masyarakat. Selain itu, jurnal penelitian ini juga diharapkan dapat menjadi sumber ide ataupun inspirasi bagi para praktisi yang berkecimpung di bidang komunikasi dan informatika serta menjadi sumber dokumentasi ilmiah bidang komunikasi dan informatika.

Dewan Redaksi akan terus berusaha meningkatkan mutu jurnal sehingga nantinya dapat menjadi acuan bagi perkembangan ilmu di bidang komunikasi dan informatika. Selain itu, juga diharapkan para pembaca untuk dapat ikut memberikan kontribusi dengan mengirimkan artikel ilmiah untuk jurnal ini.

--Redaksi--

Pelindung:

Rektor Universitas Wijaya Kusuma Surabaya

Penanggung Jawab:

Ketua Program Studi Teknik Informatika
Fakultas Teknik - Universitas Wijaya Kusuma
Surabaya

Penyunting Ahli:

Basuki Widodo
Surya Sumpena
Roni

Tim Penyunting

Ketua:

Nia Saurina

Anggota:

Okky Tri Hutomo
Tjatur Sari Widiartin
FX. Wisnu Yudho Untoro
Slamet Budi Parayitno
Nonot Wisnu Karyanto
Anang Kukuh Adisusilo

Alamat Redaksi:

Program Studi Teknik Informatika
Fakultas Teknik-UWKS
Jl. Dukuh Kupang XXV/54 Surabaya
Telp:031-5677577
Fax:031-5920741

Email:info@melekit.tif.ft-uwks.ac.id
Website:www.melekit.tif.ft-uwks.ac.id

Redaksi menerima sumbangan tulisan yang belum pernah diterbitkan dalam media lain dengan format dan aturan seperti pada halaman belakang jurnal ini. Naskah yang masuk dievaluasi dan disunting untuk keseragaman format, istilah dan lain-lain.



Daftar Isi

- (1) **APLIKASI KAMUS BAHASA INDONESIA-JAWA MENGGUNAKAN SEQUENTIAL SEARCHING PADA PLATFORM ANDROID.** Erfan Rachmad Santosa, Nia Saurina (Hal. 1 - 16)
- (2) **PERANCANGAN MODEL TATA KELOLA TEKNOLOGI INFORMASI PADA FAKULTAS TEKNIK UNIVERSITAS WIJAYA KUSUMA SURABAYA BERBASIS COBIT FRAMEWORK DOMAIN PLANNING AND ORGANIZING.** Emmy Wahyuningtyas (Hal. 17 – 26)
- (3) **APLIKASI PENGHITUNGAN PAJAK PENGHASILAN BERBASIS ANDROID DENGAN STUDI KASUS RSUD BHAKTI DHARMA HUSADA SURABAYA.** Fajar Wahyudi, Anang Kukuh Adisusilo (Hal. 27 – 42)
- (4) **MEMBANGUN SERVER VOIP TRIXBOX DENGAN CLIENT ANDROID.** Muhammad Faizal , Tjatur Widiartin. (Hal. 43 - 54)
- (5) **MENINGKATKAN KINERJA METODE DETEKSI TEPI FUZZY MENGGUNAKAN HIERARCHICAL CLUSTER ANALYSIS.** Saiful Nur Budiman, Yanuar Risah Prayogi, Stendy B. Sakur. (Hal. 55 - 60)
- (6) **VIDEO COMPOSITING MENGGUNAKAN POISSON BLENDING.** Saiful Yahya, Mochamad Hariadi, Ahmad Zaini, Eko Mulyanto Yuniarto. (Hal. 61 - 66)
- (7) **PERANCANGAN PERMAINAN BERBASIS ANDROID SEBAGAI MEDIA BELAJAR UNTUK ANAK USIA DINI UMUR 2-4 TAHUN MENGGUNAKAN AUGMENTED REALITY.** Anang Kukuh.A. (Hal. 67 - 76)
- (8) **PERGERAKAN KERUMUNAN MENGGUNAKAN RVO (RECIPROCAL VELOCITY OBSTACLE) WITH EXTENDED ORCA (OPTIMAL RECIPROCAL COLLISION AVOIDANCE) UNTUK NPC (NON PLAYER CHARACTER).** Fariz Djunaidi, Mochamad Hariadi, Supeno mardi. (Hal. 77 - 84)
- (9) **SEGMENTASI SEL DARAH PUTIH BERDASARKAN WARNA MENGGUNAKAN K-MEANS DAN OPERASI MORFOLOGI.** Shofiya Syidada, Nanik Suciati, Chastine Fatichah (Hal. 85 - 92)
- (10) **SURVEY PADA METODE COLOR CONSTANCY UNTUK PERBAIKAN WARNA CITRA DIPSTICK URINE.** Ali S. Kholimi, dan R.V. Hari Ginardi. (Hal. 93 - 98)

(9)

SEGMENTASI SEL DARAH PUTIH BERDASARKAN WARNA MENGGUNAKAN K-MEANS DAN OPERASI MORFOLOGI

Shofiya Syidada[#], Nanik Suciati^{*}, Chastine Fatichah^{*}

[#]Teknik Informatika, Universitas Wijaya Kusuma Surabaya
Jl. Dukuh Kupang XXV/54, Surabaya

shofiya@uwks.ac.id

^{*}Teknik Informatika,

Institut Teknologi Sepuluh Nopember Surabaya

nanik.suciati@its.ac.id

chastine.fatichah@cs.its.ac.id

ABSTRAK

Pada bidang medis, analisis citra mikroskopis sel darah merupakan suatu hal yang penting untuk mendiagnosa penyakit yang sedang diderita oleh pasien seperti leukemia. Metode-metode segmentasi citra sel leukemia sebelumnya mampu melakukan segmentasi sel darah putih pada citra mikroskopis sel darah putih. Namun metode-metode segmentasi tersebut belum fleksibel terhadap adanya perbedaan warna akibat perbedaan kondisi saat proses *staining*. Oleh karena itu, pada penelitian ini akan dikembangkan metode segmentasi sel darah putih leukemia secara otomatis yang fleksibel terhadap variasi warna citra. Metode usulan ini melakukan *clustering* warna menggunakan *K-means* dengan penentuan area kluster berdasarkan informasi *region of interest* sel darah putih dari hasil operasi morfologi. Data uji yang digunakan adalah 20 citra mikroskopis leukemia dengan variasi warna yang berbeda. Berdasarkan hasil uji coba yang telah dilakukan, segmentasi sel darah putih dengan variasi warna citra yang berbeda secara lebih baik dengan nilai rata-rata akurasi, *True Positive Rate* (TPR) dan *False Positive Rate* (FPR) adalah 93.07%, 84.14% dan 2.86%

Kata Kunci : *clustering*, *K-means*, operasi morfologi, *region of interest*, segmentasi sel darah putih

I. PENDAHULUAN

Pada citra mikroskopis sel darah, masing-masing obyek mempunyai perbedaan warna secara visual. Hal ini disebabkan oleh proses *staining*, yaitu pemberian zat warna, yang dilakukan sebelum pengamatan pada mikroskop dilakukan. Proses *staining* dilakukan untuk mendapatkan visualisasi yang baik pada bagian-bagian sel yang akan diamati. Perbedaan kondisi pada proses ini, seperti konsentrasi, suhu dan lama pewarnaan, akan menyebabkan variasi informasi warna pada citra sel darah [1].

Citra sel leukemia berwarna ini lebih kaya informasi, sehingga segmentasi citra sel leukemia menggunakan citra berwarna akan lebih mudah dilakukan, didukung dengan adanya perbedaan warna yang jelas antara masing-masing obyek. Sedangkan pada citra grayscale, nilai intensitas area sitoplasma hampir sama dengan area sel darah merah sehingga akan sulit mendapatkan area sitoplasma secara akurat menggunakan teknik *thresholding* biasa.

Segmentasi sel dengan cara *clustering* warna berhasil dilakukan oleh Scotti [2] dan Wang,

dkk [3]. Wang, dkk melakukan *clustering* warna berdasarkan warna-warna yang dimiliki oleh sel darah merah, sel darah putih dan keping darah menggunakan *fuzzy c-means*. Sedangkan Scotti [2] menggunakan *clustering* warna pada ruang warna $L^*a^*b^*$ secara *unsupervised*. *Cluster* yang terbentuk dikelompokkan berdasarkan persamaan warna. Untuk menemukan *cluster* dari sel leukemia digunakan citra referensi yang menunjukkan posisi sel leukemia. Citra referensi diperoleh dengan melakukan *thresholding* pada layer b^* yang diasumsikan mengandung warna area sel yaitu warna biru. Berdasarkan uji coba yang telah dilakukan, kedua metode ini bisa melakukan segmentasi sel darah secara efektif. Namun kedua metode ini memerlukan informasi awal berupa data warna masing-masing obyek. Selain itu variasi warna citra akibat proses *staining* tidak dapat diatasi oleh metode ini.

Untuk menemukan lokasi sel leukemia tanpa perlu melakukan inisialisasi warna, Madhloom melakukan segmentasi dan lokalisasi sel leukemia dengan memanfaatkan fitur warna citra sel darah pada ruang warna HSV dan melakukan rekonstruksi

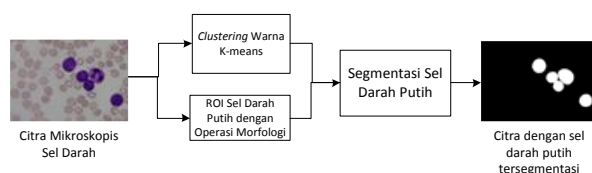
morfologi untuk mendapatkan area sel leukemia [4]. Metode yang digunakan cukup efektif dalam menemukan lokasi atau *region of interest* (ROI) sel leukemia.

Berdasarkan penelitian tentang segmentasi sel darah putih sebelumnya, untuk mendapatkan hasil segmentasi yang akurat diperlukan inisialisasi warna dari komponen sel darah yang ada pada citra secara manual. Padahal perbedaan proses *staining* menyebabkan variasi warna pada citra mikroskopis sel darah yang dihasilkan. Adanya variasi warna ini menyebabkan inisialisasi warna harus dilakukan untuk setiap citra.

Oleh karena itu, pada penelitian ini akan diusulkan sebuah metode segmentasi citra sel leukemia berwarna secara otomatis yang fleksibel terhadap variasi warna mendapatkan area sel leukemia. Metode ini merupakan kombinasi dari metode *clustering* warna menggunakan *K-means* dimana penentuan *cluster* dilakukan dengan menemukan ROI melalui operasi morfologi.

II. METODOLOGI PENELITIAN

Pada penelitian ini dilakukan segmentasi untuk mendapatkan area sel darah putih. Rancangan metodologi penelitian yang dilakukan ditunjukkan oleh Gambar 1.



Gambar 1. Metodologi Penelitian Metode Usulan

A. Citra Mikroskopis Leukemia

Citra mikroskopis sel darah merupakan visualisasi dari sel-sel yang ada pada darah. Citra sel darah berisi sel darah merah, sel darah putih, keping darah, dan *background*. Proses pengambilan (*capture*) gambar dilakukan dengan menggunakan kamera yang dipasang pada mikroskop optik.

Citra sel darah diambil dari sampel darah, yaitu setetes darah yang telah dikeringkan (*smear*). Setelah itu dilakukan proses *staining* yaitu pemberian zat warna pada sampel darah. Proses ini bertujuan untuk mendapatkan visualisasi sel atau bagian dari sel yang lebih baik pada pengamatan menggunakan mikroskop.

Hal ini dilakukan karena sel darah merah dan sel darah putih mempunyai warna yang transparan. Pemberian zat warna akan memperjelas perbedaan masing-masing komponen sel darah. Akan tetapi, perbedaan kondisi pada proses *staining* ini, seperti

konsentrasi, suhu dan lama pewanaan, akan menyebabkan variasi warna citra mikroskopis sel darah yang diperoleh [1].

B. Clustering Warna K-means

Tahap pertama proses segmentasi adalah melakukan *clustering* warna. Sebelum proses *clustering*, dilakukan pra-proses yaitu melakukan konversi warna. Citra pada ruang warna RGB dikonversi ke ruang warna $L^*a^*b^*$. Hal ini dilakukan untuk mengurangi pengaruh dari iluminasi citra disebabkan oleh konsentrasi darah serta kondisi pada proses *staining* yang berbeda [2]. Selain itu, untuk mengurangi dimensi warna menjadi hanya dua dimensi saja bila dibandingkan dengan ruang warna RGB yang tiga dimensi [5].

Clustering warna dilakukan secara *unsupervised* menggunakan *k-means*. Jumlah *cluster* ditentukan berdasarkan persepsi visual manusia terhadap citra mikroskopis sel darah secara umum. Penampakan warna citra mikroskopis sel darah secara umum terbagi menjadi warna inti sel, warna sitoplasma, warna sel darah merah dan warna *background*. Selain itu menurut penelitian yang dilakukan oleh Wang jumlah *cluster* warna pada citra sel darah dibagi menjadi 6 *cluster* warna yaitu 2 variasi warna untuk sel darah merah, 3 variasi warna untuk inti sel dan satu warna sitoplasma [3]. Oleh karena itu pada proses *clustering* warna ini akan dibagi menjadi 6 *cluster* warna.

C. Operasi Morfologi untuk Menemukan Region of Interest Sel Darah Putih

Pencarian ROI obyek-obyek pada citra mikroskopis sel darah dilakukan dengan melakukan konversi warna dari RGB ke HSV. Ruang warna HSV digunakan karena bebas dari pengaruh pencahayaan pada citra [1]. Demikian halnya penelitian yang dilakukan oleh Madhloom menggunakan layer *H* dan *S* yang digunakan sebagai citra *mask* dan *marker* dalam rekonstruksi citra untuk mendapatkan area sel darah putih [4].

Pada layer *S*, area inti sel darah putih tampak lebih terang daripada area yang lain. Oleh karena itu pengolahan layer *S* ini digunakan untuk menandai posisi inti sel darah putih. Pada layer *H*, memperlihatkan semua obyek yang ada pada citra. Oleh karena itu, penentuan area sel darah merah bisa dilakukan dengan menggunakan layer *H*. Sedangkan area sel darah putih diperoleh dengan menggunakan kombinasi layer *S* dan *H* dengan melakukan rekonstruksi morfologi.

D. Segmentasi Sel Darah Putih

Secara umum proses segmentasi sel darah putih dilakukan dengan mencocokkan posisi ROI sel darah putih dengan *cluster* warna yang diperoleh menggunakan K-means. Posisi ROI sel darah putih yang telah diperoleh digunakan sebagai informasi untuk menentukan *cluster* yang bersesuaian. Posisi sel darah putih dibandingkan dengan semua hasil *cluster k-means*. Bila posisi sel darah putih ada yang beririsan dengan satu atau lebih area *cluster* maka *cluster* tersebut merupakan area dari sel darah putih.

Area inti sel merupakan bagian dari area sel darah putih. Dengan demikian, proses pemetaan ROI obyek pada *cluster* warna dilakukan dalam 2 tahap, yaitu pencarian *cluster* warna untuk area inti sel dan *cluster* warna untuk sel darah putih secara keseluruhan.

Pencarian *cluster* warna untuk area inti sel menggunakan ROI inti sel sebagai informasi awal untuk menentukan *cluster* warna dari inti sel. ROI inti sel digunakan untuk menghitung luas area obyek inti sel yang dinyatakan dengan L_{inti} . L_{inti} ini digunakan untuk mendapatkan nilai *threshold* jumlah piksel minimal yang beririsan dengan masing-masing *cluster*. *Cluster* yang memenuhi syarat akan menjadi kandidat *cluster* warna dari inti sel.

Langkah selanjutnya adalah mencari irisan antara ROI inti sel dengan semua *cluster* warna. Bila diketahui jumlah seluruh *cluster* warna adalah n , *cluster* warna dinyatakan dengan $cluster\{i\}$ dimana $i = 1, 2, \dots, n$ dan ROI inti sel dinyatakan dengan ROI_{inti} maka area *cluster* inti sel adalah *cluster* yang mempunyai irisan dengan ROI inti sel dan jumlah piksel yang beririsan lebih besar dari nilai *threshold* yang ditetapkan. Perhitungan ini menggunakan persamaan 1 dan 2.

$$ROI_{inti} \cap cluster\{i\} \neq \emptyset, \quad (1)$$

$$n(ROI_{inti} \cap cluster\{i\}) \geq 10\% * L_{inti}, \quad (2)$$

dimana nilai *threshold* yang ditetapkan adalah 10% dari luas obyek inti sel.

Apabila inti sel mempunyai *cluster* warna lebih dari satu atau berjumlah k maka dilakukan uji kedekatan atau ketetangaan antar *cluster* inti sel. $Cluster_{inti}$ adalah *cluster* warna inti sel dan mempunyai pusat *cluster* yang dinyatakan dengan

$$Cluster_{inti}\{x\} = (a_x, b_x), \quad (3)$$

dimana $Cluster_{inti}\{x\}$ adalah pusat *cluster* inti ke- x dengan x dan y adalah bilangan bulat positif dengan nilai $1, 2, \dots, k$, a_x adalah nilai warna pada layer a^* dari pusat *cluster* dan b_x adalah nilai warna pusat *cluster* pada layer b^* . Uji kedekatan ini dilakukan dengan menghitung jarak antar pusat *cluster* menggunakan rumus *euclidian distance*.

$$d(Cluster_{inti}\{x\}, Cluster_{inti}\{y\}) = \sqrt{(a_x - a_y)^2 + (b_x - b_y)^2}, \quad (4)$$

Dua *cluster* warna dikatakan berdekatan atau bertetangga bila jarak pusat *cluster* minimal kurang dari nilai median dari semua jarak *cluster* inti ke seluruh *cluster* lain, dinyatakan oleh persamaan

$$d(Cluster_{inti}\{x\}, Cluster_{inti}\{y\}) < \text{median}(d(Cluster_{inti}\{1, \dots, k\})) \quad (5)$$

Untuk mempertegas kedekatan kedua *cluster* dihitung selisih antara median dengan jarak bernilai lebih dari *threshold*, menggunakan persamaan

$$\text{median} - d(Cluster_{inti}\{i\}) > \text{threshold}, \quad (6)$$

dimana nilai *threshold* yang diambil adalah 5.

Selanjutnya adalah melakukan evaluasi terhadap seluruh kandidat *cluster* inti yang telah diperoleh. Evaluasi dilakukan dengan memperhitungkan jumlah area yang beririsan dari kandidat *cluster* inti dengan ROI inti. Kandidat *cluster* yang mempunyai jumlah area beririsan paling banyak akan menjadi *cluster* inti.

Setelah *cluster* warna inti sel diperoleh maka langkah selanjutnya adalah pencarian *cluster* warna untuk area sel darah putih. ROI sel darah putih dan *cluster* warna inti merupakan informasi awal yang digunakan untuk menentukan *cluster* warna dari sel. *Cluster* warna inti yang telah diperoleh otomatis akan menjadi *cluster* warna sel darah putih. *Cluster* warna lainnya akan dipertimbangkan sebagai kandidat *cluster* warna sel dengan memperhatikan kedekatan *cluster* warna tersebut dengan *cluster* warna inti yang telah terbentuk menggunakan persamaan 5 dan 6. Selain itu irisan *cluster* warna dengan ROI sel darah putih juga dipertimbangkan sebagai syarat untuk menjadi kandidat *cluster* warna sel.

E. Evaluasi Performa

Evaluasi kinerja dari algoritma segmentasi sel darah putih leukemia yang diusulkan menggunakan nilai akurasi, TPR (*True Positive Rate*), dan FPR (*False Positive Rate*). Akurasi, TPR dan FPR diperoleh dengan menggunakan *confusion matrix*, sebagaimana yang ditunjukkan oleh Gambar 2. Dalam matriks ini dihitung empat nilai yaitu TP (*True Positive*), FP (*False Positive*), TN (*True Negative*) dan FN (*False Negative*). TP adalah jumlah piksel obyek yang berhasil disegmentasi dan sesuai dengan obyek pada citra *ground truth*. FP adalah jumlah piksel *background* pada citra *ground truth* yang dikenali sebagai piksel obyek. FN adalah jumlah piksel obyek pada citra *ground truth* yang tidak berhasil disegmentasi. TN adalah jumlah piksel

background yang berhasil disegmentasi dan sesuai dengan *background* pada citra *ground truth*.

		Ground Truth	
		T	F
Hasil segmentasi	T	TP (<i>True Positive</i>)	FP (<i>False Positive</i>)
	F	FN (<i>False Negative</i>)	TN (<i>True Negative</i>)

Gambar 2. *Confusion Matrix*

Nilai TPR dan FPR diperoleh dari dengan menggunakan rumus 7 dan 8 berikut ini.

$$TPR = \frac{TP}{TP+FN} \quad (7)$$

$$FPR = \frac{FP}{FP+TN} \quad (8)$$

Sedangkan nilai akurasi hasil segmentasi dihitung dengan menggunakan rumus

$$Akurasi = \frac{TP+TN}{TP+FP+TN+FN} \quad (9)$$

dimana TP adalah jumlah piksel TP, TN adalah jumlah piksel TN, FP adalah jumlah piksel FP dan FN adalah jumlah piksel FN. Nilai TPR, FPR dan akurasi diperoleh dengan membandingkan citra hasil segmentasi sistem yang diusulkan dengan citra *ground truth*, yaitu citra yang disegmentasi secara manual.

III. HASIL DAN PEMBAHASAN

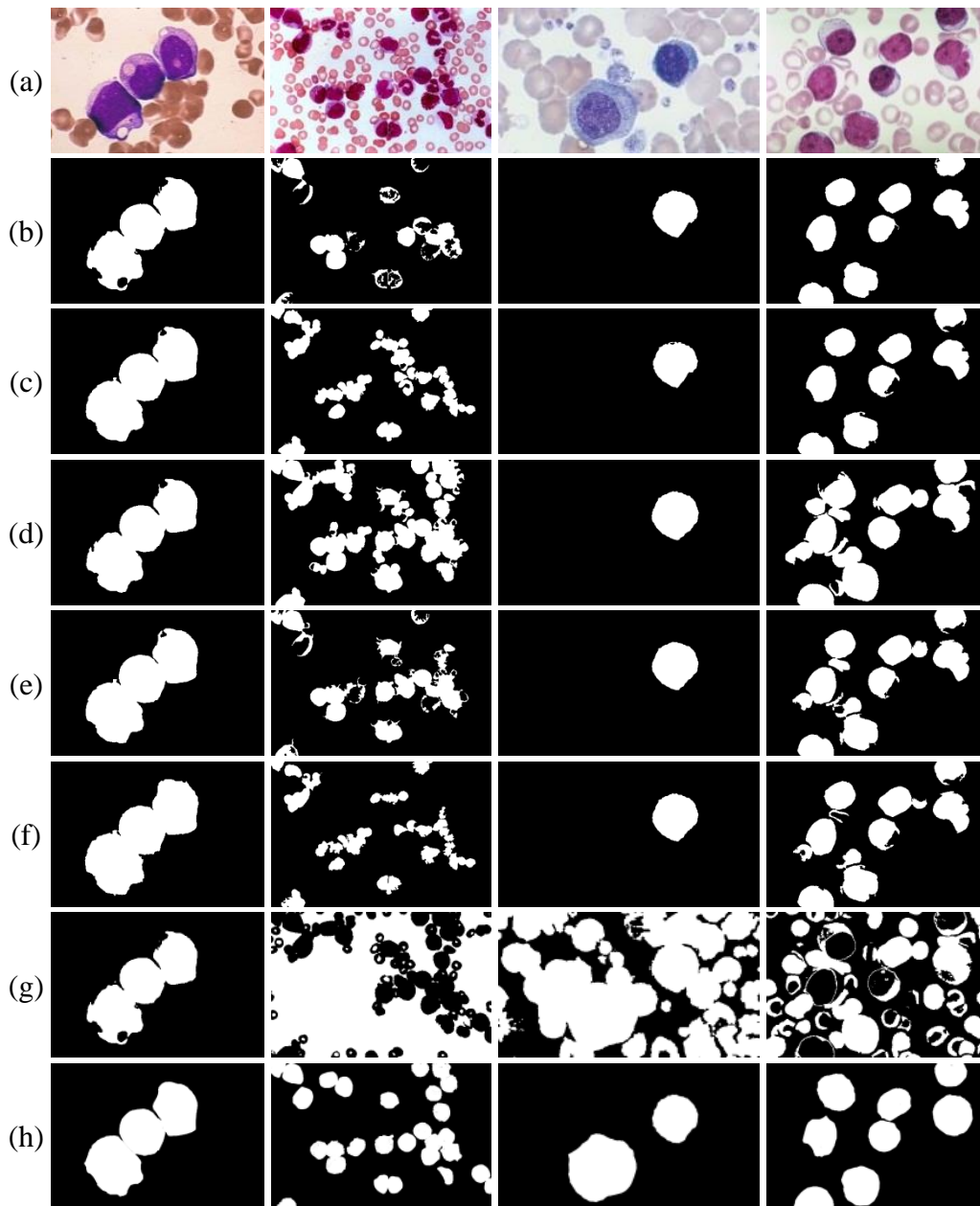
Metode usulan ini melakukan segmentasi sel darah putih yang fleksibel terhadap variasi warna citra mikroskopis leukemia. Metode ini bertujuan untuk mendapatkan hasil segmentasi sel darah putih tanpa harus melakukan inisialisasi warna obyek-obyek yang ada pada citra. Sedangkan metode Scotti pada [2] melakukan segmentasi sel darah putih dengan inisialisasi warna sel darah putih adalah biru. Hasil dari perbandingan kedua metode ini akan menunjukkan kontribusi metode usulan pada penelitian ini.

Untuk melakukan perbandingan secara seimbang, pengujian dilakukan dengan menggunakan data yang sama yaitu 20 citra mikroskopis leukemia dengan variasi warna yang berbeda. Jumlah *cluster* warna yang digunakan pada metode usulan adalah 3, 6, 9, 12 dan 15 *cluster*.

Contoh hasil uji coba sel darah putih yang disegmentasi dengan metode usulan ditunjukkan oleh Gambar 3 dan performa hasilnya ditunjukkan pada Tabel 1. Sedangkan nilai rata-rata akurasi, TPR dan FPR hasil metode usulan dengan kombinasi jumlah *cluster* 3, 6, 9, 12 dan 15 *cluster* ditunjukkan oleh Tabel 2. Tabel perbandingan rata-rata akurasi,

TPR dan FPR hasil segmentasi sel darah putih metode usulan dengan metode Scotti ditunjukkan oleh Tabel 3.

Pada Gambar 3 menunjukkan bahwa citra yang mempunyai sel darah putih berwarna dominan biru adalah citra pertama dan citra ketiga. Pada citra pertama sel darah putih berwarna biru-ungu dan sel darah merah berwarna merah muda. Sedangkan pada citra ketiga sel darah putih dan sel darah merah mempunyai warna dominan biru. Citra kedua dan keempat mempunyai sel darah putih berwarna dominan merah.



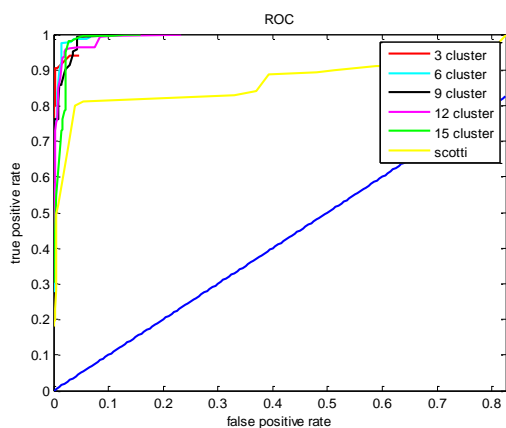
Gambar 3. Hasil Segmentasi Sel Darah Putih Menggunakan Variasi Jumlah *Cluster* Warna. (a) Citra Asli; (b) 3; (c) 6; (d) 9; (e) 12; (f) 15; (g) Hasil Segmentasi Sel Darah Putih Metode Scotti; (h) *Ground Truth*.

Hasil segmentasi metode usulan dapat mendapatkan area sel darah putih untuk citra dengan variasi warna. Sedangkan hasil segmentasi metode Scotti dengan inisialisasi warna biru untuk sel darah putih. Performa hasil segmentasi dari Gambar 3 yang ditampilkan pada Tabel 1.

Tabel 1 menyajikan performa hasil segmentasi metode usulan pada citra dengan variasi warna dari Gambar 3. Performa terbaik metode usulan untuk citra pertama diperoleh

dengan menggunakan jumlah *cluster* warna 15. Untuk citra kedua dan ketiga, performa terbaik diperoleh dengan menggunakan jumlah *cluster* warna 9. Pada citra keempat performa terbaik diperoleh dengan menggunakan jumlah *cluster* warna 6. Sedangkan hasil segmentasi dengan inisialisasi warna biru sebagai warna sel darah putih menggunakan metode Scotti [2] memperlihatkan performa terbaik dimiliki oleh citra pertama karena mempunyai sel darah putih

yang berwarna biru-ungu. Akan tetapi untuk citra ketiga, sel darah merah turut terseleksi karena mempunyai warna biru. Oleh karena itu, dari keempat citra performa hasil segmentasi metode usulan mempunyai performa yang lebih baik.



Gambar 4. Performa Hasil Segmentasi Menggunakan Metode Usulan dengan Variasi Jumlah *Cluster* Warna Dibandingkan Hasil Segmentasi Menggunakan Metode [2].

TABEL 1

PERFORMA HASIL SEGMENTASI DARI GAMBAR 3

Jumlah <i>Cluster</i>	Performa	Citra ke-			
		1	2	3	4
3	Akurasi	98.13	86.66	90.25	93.10
	TPR	90.40	41.64	28.50	71.68
	FPR	0.02	0.84	0.00	0.02
6	Akurasi	99.40	82.33	90.11	93.91
	TPR	98.60	43.33	27.49	75.23
	FPR	0.41	6.84	0.00	0.09
9	Akurasi	99.00	89.88	90.56	89.86
	TPR	95.14	91.33	31.36	85.80
	FPR	0.08	10.52	0.08	8.84
12	Akurasi	99.34	88.58	90.55	90.70
	TPR	99.16	58.30	30.86	75.40
	FPR	0.61	3.01	0.02	4.39
15	Akurasi	99.00	81.68	90.49	92.08
	TPR	99.88	35.46	30.35	76.19
	FPR	1.21	5.49	0.01	2.81
Scotti	Akurasi	98.02	18.63	49.09	63.38
	TPR	89.84	22.83	99.70	52.34
	FPR	0.02	82.54	58.90	33.08

Pada Tabel 2 terlihat bahwa rata-rata akurasi dan TPR tertinggi yaitu 93.78% dan 82.87% dicapai saat menggunakan 9 *cluster* warna.

Sedangkan nilai FPR terendah adalah 0.50% dicapai saat menggunakan *cluster* 3.

Perbandingan rata-rata nilai akurasi, TPR dan FPR hasil pengujian menggunakan metode usulan dengan metode Scotti [2] ditampilkan oleh Tabel 3. Pada tabel tersebut terlihat bahwa rata-rata akurasi dan TPR metode usulan yaitu 92.20% dan 78.34% lebih baik bila dibandingkan dengan rata-rata akurasi dan TPR penggunaan metode Scotti, yaitu 69,01% dan 67.51%. Sedangkan nilai FPR metode usulan adalah 2.59% lebih rendah bila dibandingkan FPR metode Scotti yaitu 29.79%.

Perbandingan performa TPR dengan FPR menggunakan variasi jumlah *cluster* digambarkan menggunakan kurva ROC seperti yang ditunjukkan oleh Gambar 4. Performa metode usulan yang paling baik adalah performa penggunaan jumlah *cluster* warna yang nilai TPR-nya paling cepat mendekati nilai 1. Pada Gambar 4, nilai TPR yang paling cepat mendekati angka 1 adalah saat penggunaan 9 *cluster* warna, yaitu garis warna hitam. Oleh karena itu, dapat disimpulkan bahwa performa terbaik metode usulan dapat dicapai pada saat penggunaan 9 *cluster* warna berdasarkan hasil pada Tabel 2 serta penampakan dari kurva ROC.

TABEL 2

RATA-RATA AKURASI, TPR DAN FPR METODE USULAN MENGGUNAKAN VARIASI JUMLAH *CLUSTER*

Jumlah <i>Cluster</i>	Akurasi	TPR	FPR
3	91.06	68.91	0.50
6	91.87	75.65	1.60
9	93.78	82.87	3.02
12	92.37	82.64	3.67
15	91.93	81.62	4.16

TABEL 3

PERBANDINGAN RATA-RATA AKURASI, TRP DAN FPR METODE USULAN DENGAN METODE SCOTTI

Metode	Akurasi	TPR	FPR
Scotti [2]	69.01	67.51	29.79
Usulan	92.20	78.34	2.59

IV. KESIMPULAN

Penelitian ini telah menjelaskan tentang metode segmentasi sel darah putih secara otomatis pada citra mikroskopis leukemia yang fleksibel terhadap variasi warna citra. Segmentasi sel darah putih dilakukan menggunakan *clustering* warna *k-means* dan operasi morfologi untuk mendapatkan ROI sel darah putih.

Berdasarkan hasil uji coba yang telah dilakukan, performa terbaik metode usulan untuk

segmentasi sel darah putih dapat dicapai dengan menggunakan 9 *cluster* warna dengan nilai rata-rata akurasi, TPR dan FPR adalah 93.78%, 82.87% dan 3.02%

Metode usulan dapat melakukan segmentasi sel darah putih lebih baik pada citra mikroskopis leukemia dengan variasi warna yang berbeda tanpa perlu melakukan inisialisasi warna terlebih dahulu dibandingkan dengan metode Scotti [2]. Performa rata-rata akurasi dan TPR metode usulan yaitu 92.20% dan 78.34% lebih baik bila dibandingkan dengan rata-rata akurasi dan TPR metode Scotti, yaitu 69,01% dan 67.51%. Sedangkan nilai FPR metode usulan adalah 2.59% lebih rendah bila dibandingkan FPR metode Scotti yaitu 29.79%.

Penelitian ini menggunakan metode *hard-clustering* yaitu *k-means* untuk melakukan *clustering* warna pada citra mikroskopis leukemia, sehingga warna dikelompokkan secara tegas. Penelitian lebih lanjut dapat menggunakan metode *soft-clustering* seperti *fuzzy c-means* untuk memperbaiki hasil *cluster* warna yang terbentuk.

V. DAFTAR PUSTAKA

- [1] Huang, D. C., Hung, K. D., & Chan, Y. K. "A computer assisted method for leukocyte nucleus segmentation and recognition in blood smear images". *Journal of Systems and Software*, 85(9), pp. 2104-2118, 2012.
- [2] Scotti, F. "Robust segmentation and measurements techniques of white cells in blood microscope images". *Instrumentation and Measurement Technology Conference*, pp. 43-48. IEEE, 2006.
- [3] Wang, E. Y., Gou, Z., Miao, A. M., Peng, S. Q., Niu, Z. Y., & Shi, X. L. "Recognition of blood cell images based on color fuzzy clustering". *Fuzzy Information and Engineering*, Volume 2, pp. 69-75. Springer Berlin Heidelberg, 2009.
- [4] Madhloom, H. T., Kareem, S. A., & Ariffin, H.. "An image processing application for the localization and segmentation of lymphoblast cell using peripheral blood images". *Journal of medical systems*, 36(4), pp. 2149-2158, 2012.
- [5] Mohapatra, S., Samanta, S. S., Patra, D., & Satpathi, S. (2011). "Fuzzy based blood image segmentation for automated leukemia detection". *Devices and Communications (ICDeCom)*, International Conference (pp. 1-5). IEEE.

{Halaman sengaja dikosongkan}
{Jurnal **Melek IT**}

(1)
APLIKASI KAMUS BAHASA INDONESIA-JAWA MENGGUNAKAN *SEQUENTIAL SEARCHING* PADA PLATFORM ANDROID

Erfan Rachmad Santosa¹, Nia Saurina²

Program Studi Teknik Informatika, Universitas Wijaya Kusuma Surabaya, Jl. Dukuh Kupang XXV/54
teajus@yahoo.co.id , niasaurina@yahoo.com

ABSTRAK

Bahasa merupakan hal mendasar yang sangat dibutuhkan setiap orang dalam berkomunikasi dan bertukar informasi. Terdapat berbagai ragam bahasa di Indonesia. Salah satunya yaitu Bahasa Jawa yang merupakan bahasa daerah yang sebagian besar digunakan sebagai bahasa pengantar sehari-hari di wilayah provinsi Jawa Timur. Namun seiring perkembangan zaman, tidak sedikit yang kurang memahami hal tentang kosakata bahasa Jawa terutama masyarakat Jawa generasi saat ini yang mulai terpengaruh globalisasi.

Aplikasi kamus dibangun mengikuti perkembangan teknologi *mobile* saat ini, yaitu teknologi yang berbasis Android. Dengan pendekatan teknologi diharapkan informasi mengenai itu semua dapat tersampaikan pada masyarakat luas dengan tujuan agar aplikasi kamus tersebut dapat difungsikan semaksimal mungkin.

Pada Tugas Akhir ini akan dijelaskan aplikasi kamus bahasa Indonesia-Jawa menggunakan *sequential searching* pada platform Android. Dengan adanya hasil ini nantinya dapat diharapkan dapat memberi kemudahan dalam mendapatkan informasi khususnya mengenai kosakata bahasa Jawa dalam tingkat tutur *ngoko*, *krama*, dan *krama inggil*. Aplikasi kamus *mobile* ini juga dapat dijadikan media pembelajaran untuk memahami istilah atau arti kata bahasa Jawa sehingga menambah pengetahuan dan wawasan.

Kata kunci : Aplikasi Kamus Mobile, Bahasa Jawa, Media Pembelajaran, Platform Android,

PENDAHULUAN

Latar Belakang

Perkembangan teknologi semakin pesat dan cepat, khususnya teknologi informasi dan komunikasi. Dengan perkembangan teknologi yang kian maju, manusia dapat membuat berbagai macam peralatan sebagai alat bantu dalam menjalankan berbagai aktivitas untuk mendukung produktifitas. Seiring dengan tingkat mobilitas yang tinggi, beberapa tahun terakhir tengah marak perangkat bergerak atau *mobile device*. Salah satu perangkat *mobile* yang paling pesat adalah *Handphone* dimana hampir setiap orang memilikinya. Berbagai macam fitur telah ditanamkan, seperti pengolah gambar dan video, pengolah dokumen dan lain sebagainya. Hal ini tak lepas dari penggunaan Sistem Operasi pada *handphone*. Layaknya pada komputer, *handphone* pun dapat di *installasi* berbagai macam aplikasi yang diinginkan.

Bahasa Jawa adalah bahasa yang digunakan oleh masyarakat terutama di Jawa Tengah, Jawa Timur, dan sebagian daerah Banten. Pembentukan bahasa Jawa dipengaruhi oleh berbagai faktor, seperti umur, dialek, golongan, berbagai derajat kemurnian, dan perbedaan sosial.

Bahasa Jawa dikenal sebagai bahasa yang rumit karena memiliki berbagai macam variasi sesuai tingkatannya, yang disebut tingkat tutur

(*speech level*). Tingkat tutur dalam bahasa Jawa merupakan suatu sistem untuk menunjukkan derajat formalitas dan derajat hormat oleh si pembicara kepada orang yang diajak bicara. Semakin besar derajat hormat dan formalitas dalam ujaran semakin besar pula derajat kesopanan yang ditunjukkannya.

Saat ini penggunaan bahasa Jawa kurang populer dikalangan masyarakat jawa itu sendiri. Kurangnya minat generasi muda terhadap bahasa Jawa disebabkan karena bahasa Jawa dianggap tidak dapat digunakan untuk meniti karir atau mencari kerja, sulit, bahkan ada yang menganggap bahasa Jawa sebagai bahasa *wong desa*, sehingga mereka merasa malu untuk menggunakan Bahasa Jawa. Sebagai akibatnya, unggah-ungguh dalam bahasa Jawa juga semakin dilupakan. Bahasa Jawa yang dikenal anak-anak usia sekolah kebanyakan hanya logat *ngoko* (untuk percakapan sehari-hari), bahkan ketika mereka berbicara dengan orang yang lebih tua. Bahasa Jawa halus (*krama alus*) semakin jarang digunakan, yang menyebabkan terkikisnya dari nilai unggah-ungguh bahasa Jawa.

Android sebagai sistem operasi yang dapat digunakan di berbagai perangkat *mobile*. Android memiliki tujuan utama untuk memajukan inovasi piranti telepon bergerak agar pengguna mampu mengeksplorasi kemampuan dan menambah pengalaman lebih dibandingkan dengan platform *mobile* lainnya. Hingga saat ini Android terus

berkembang, baik secara sistem maupun aplikasinya.

Untuk itu diperlukan adanya aplikasi kamus bahasa Jawa berplatform android yang dapat digunakan sebagai pencarian kosakata. Dengan Adanya aplikasi kamus bahasa Jawa ini, diharapkan bisa memotivasi masyarakat dan pelajar untuk mempelajari atau memelihara bahasa Jawa.

Tujuan

Tujuan Tugas Akhir ini adalah sebagai berikut :

1. Menarik minat masyarakat untuk mempelajari bahasa Jawa.
2. Memudahkan dalam pencarian istilah-istilah yang belum diketahui artinya tanpa harus membuka buku terlebih dahulu.

TINJAUAN PUSTAKA

Pengertian Kamus

Kamus adalah sejenis buku rujukan yang menerangkan makna kata-kata. Ia berfungsi untuk membantu seseorang mengenal perkataan baru. Selain menerangkan maksud kata, kamus juga mungkin mempunyai pedoman sebutan, asal-usul (*etimologi*) sesuatu perkataan dan juga contoh penggunaan bagi sesuatu perkataan. Untuk memperjelas kadang kala terdapat juga ilustrasi di dalam kamus. yang tidak terhingga dalam dan luasnya. Dewasa ini kamus merupakan khazanah yang memuat perbendaharaan kata suatu bahasa, yang secara ideal tidak terbatas jumlahnya. Setiap kebudayaan besar di dunia bangga akan kamus bahasanya. Dalam kenyataannya kamus itu tidak hanya menjadi lambang kebanggaan suatu bangsa, tetapi juga mempunyai fungsi dan manfaat praktis.

Pengertian Bahasa

Bahasa dapat didefinisikan dalam berbagai ragam tergantung dari ciri-ciri apa yang ingin ditonjolkan. Definisi tersebut dapat bersifat luas, sehingga mencakupi semua bentuk komunikasi atau secara sempit disampaikan sedemikian rupa sehingga melibatkan seperangkat kaidah bahasa saja. Berikut ini disinggung isi dua definisi bahasa, dari yang agak sederhana sampai pada yang cukup panjang. Yang dibahas isinya saja, sekedar membantu kita dalam pembahasan mengenai penerjemahan nantinya.

Ada ahli yang beranggapan bahwa 'bahasa adalah sebuah sistem simbol yang bersifat mana suka dan dengan sistem itu suatu kelompok sosial bekerja sama. Namun, ada juga ahli bahasa yang beranggapan bahwa 'bahasa adalah sebuah sistem

berstruktur mengenai bunyi dan urutan bunyi bahasa yang sifatnya mana suka, yang digunakan, atau yang dapat digunakan dalam komunikasi antar individu oleh sekelompok manusia dan secara agak tuntas memberi nama kepada benda-benda. peristiwa-peristiwa dan proses-proses dalam lingkungan hidup manusia (Carrol, 1959)

Apabila kedua isi. definisi tersebut kita bandingkan, maka dengan mudah dapat kita lihat bahwa isi definisi yang diajukan Carrol merupakan definisi yang lebih luas cakupannya dari definisi yang pertama. Demikian juga, dari isi definisi tersebut kita dapat mengambil beberapa hal penting.

Pertama, bahasa merupakan sistem yang mempunyai struktur (*structured system*) sebagaimana halnya dengan sistem lain. Bahasa memiliki pola dan berdasarkan pola itu adalah bahasa digunakan. Pola (sistem gramatikal) tersebut pada umumnya bersifat statis; perubahan mendasar jarang terjadi dan jika terjadi tentu melalui proses yang cukup lama. Karena bahasa itu memiliki pola-pola, maka bahasa merupakan sebuah sistem, dan karena adanya sistem inilah maka bahasa dapat dibandingkan, dialihkan, dipelajari dan diajarkan.

Bahasa Indonesia

Bahasa Indonesia berasal dari bahasa Melayu. Namun sekarang ini bahasa Indonesia yang kita gunakan sebagai bahasa nasional tidak lagi sama dengan bahasa asalnya, bahasa Melayu. Bahasa Melayu seperti bahasa Melayu Riau kini sama kedudukannya dengan bahasa-bahasa daerah lain di Indonesia. Bahasa Indonesia telah tumbuh dan berkembang dengan pesat, bukan lagi hanya sebagai bahasa pergaulan sehari-hari, tetapi telah tumbuh menjadi bahasa ilmiah dan teknologi. Bahasa Indonesia sebagai bahasa ilmiah digunakan untuk menulis buku, skripsi, berpidato, kuliah, dan sebagainya. Selain itu bahasa Indonesia memperkaya dirinya dengan mengambil unsur-unsur baik dari bahasa daerah maupun bahasa asing yang disesuaikan dengan sistem fonologi, morfologi, dan sintaksisnya. Penyerapan kata-kata asing itu kemudian diatur dalam buku "Pedoman Umum Ejaan Bahasa Indonesia yang Disempurnakan" dan "Pedoman Umum Pembentukan Istilah".

Penggunaan bahasa Indonesia sebagai bahasa negara memerlukan pengembangan kata dan istilah dalam berbagai bidang ilmu. Kekayaan kosakata suatu bahasa dapat menjadi indikasi kemajuan bangsa pemilik bahasa tersebut. Penggunaan kosakata bahasa asing makin meluas sejak terjadi perubahan tatanan kehidupan baru (globalisasi) yang telah mengubah pola pikir dan perilaku masyarakat pada berbagai sendi kehidupan.

Bahasa Jawa

Bahasa Jawa adalah suatu bahasa daerah yang merupakan bagian dari kebudayaan nasional Indonesia, yang hidup dan tetap dipergunakan dalam masyarakat bahasa yang bersangkutan. Bahasa Jawa mengenal tata bahasa (*undhak-undhuk basa*) dan menjadi bagian integral dalam tata krama masyarakat Jawa dalam berbahasa. Terdapat tiga bentuk utama variasi, yaitu *ngoko* (kasar), *madya* (biasa), dan *krama* (halus). Di antara masing-masing bentuk ini terdapat bentuk penghormatan dan perendahan seseorang dapat berubah-ubah registernya pada suatu saat tergantung status yang bersangkutan dan lawan bicara. Status bisa ditentukan oleh usia, posisi sosial, atau hal-hal lain. Seorang anak yang bercakap-cakap dengan sebayanya akan berbicara dengan varian *ngoko*, namun ketika bercakap dengan orang tuanya akan menggunakan *krama andhap* dan *krama inggil*. Sistem semacam ini terutama dipakai di Surakarta, Yogyakarta, dan Madiun. Dialek lainnya cenderung kurang memegang erat tata-tertib berbahasa semacam ini. Pendapat mengenai *undhak-undhuk* tingkat tutur tersebut dikemukakan oleh (Poejosoedarmo, 1973). *Undhak-undhuk* tingkat tutur bahasa Jawa terbagi atas tiga jenis yaitu : *Krama*, *Madya*, *Ngoko* dengan masing-masing sub tingkat. Berikut ini penjelasan mengenai *Undhak-undhuk* tingkat tutur tersebut :

a. Krama

Krama adalah tingkatan bahasa sangat sopan bila digunakan untuk percakapan. *Krama* juga dibagi menjadi 2 yaitu *Krama Madyo* dan *Krama Inggil*. *Krama Madya* adalah bahasa pertengahan. Sedikit di atas bahasa *Ngoko Alus* tetapi di bawah *Krama Inggil*. *Krama Madya* biasanya digunakan oleh para orang tua untuk berbicara kepada sesamanya atau kepada seseorang yang orang yang lebih mudah tetapi lebih dihormati tetapi pangkat tidak jauh beda. Seperti ketua RT dan Kepala Desa. Sedangkan *Krama Inggil* adalah bahasa paling tinggi dalam tingkatan bahasa Jawa. Bahasa ini biasa digunakan oleh “abdi dalem” kepada “gustinya”, anak kepada orang tuanya, atau rakyat biasa kepada rajanya.

b. Madya

Madya adalah bahasa yang sering digunakan dalam masyarakat pedesaan atau masyarakat gunung. *Madya* dibagi menjadi dua: *madya ngoko* dan *madya krama*. *Madya ngoko* adalah sebuah bahasa yang dikalobarasi dengan bahasa *ngoko* tetapi lebih lekat kedaerahan, bisa dibilang bahasa daerah setempat yang tak semua orang Jawa mengerti. Ciri-cirinya:

- Saya diganti menjadi kula.
- Anda menjadi dika.

- Awalan tak- diganti menjadi kula.
- Awalan ko- diganti menjadi dika.
- Awalan di- tidak berubah.

Sedangkan *madya krama* biasa digunakan masyarakat desa berbicara dengan orang yang baru kenal atau orang yang dihormati. Bisa dikatakan hampir sama dengan *ngoko andhap* tetapi memunyai batasan: orang muda kepada yang lebih tua atau dihormati. Ciri-cirinya:

- Saya, diganti menjadi kula.
- Anda, diganti menjadi sampeyan, panjenengan.
- Awalan tak- diganti menjadi kula.
- Awalan ko- diganti menjadi samang, mang.
- Akhiran -ku diganti menjadi kula.
- Akhiran -mu diganti menjadi sampéyan, samang.
- Akhiran -e tetap tidak berubah.

c. Ngoko

Ngoko adalah tingkatan bahasa paling “kasar” dalam bahasa Jawa. Bahasa *Ngoko* terbagi lagi menjadi 2 yaitu *Ngoko Lugu* dan *Ngoko Alus*. *Ngoko lugu* adalah bahasa yang paling kasar. Bahasa ini sering digunakan oleh anak-anak kecil untuk berbicara sesamanya (tidak termasuk dalam lingkungan Keraton, karena biasanya mereka menggunakan bahasa *Krama Madya*). Sedangkan *Ngoko Alus* adalah bahasa *Ngoko* yang sudah ada perpaduan dengan *Kromo*. Jadi secara kata-kata juga sudah lebih sopan. Bahasa ini sering digunakan oleh orang tua kepada anak-anaknya yang masih kecil untuk mengajari anaknya sopan santun.

d. Contoh kalimat pengguna tingkat tutur dalam bahasa jawa :

1. Saya mempunyai mangga, apa kamu mau ?
 - *Ngoko: Aku duwe pelem, apa kowe arep ?*
 - *Krama: Kula gadhah pelem, menapa sampeyang anjeng?*
 - *Krama Inggil : Kula gadhah pelem, menapa panjenengan kersa?*
2. Bu Niken disuruh menjadi guru tari di sekolah.
 - *Ngoko: Bu Niken dikon dadi guru jogged ana ing sekolahan.*
 - *Krama: Bu Niken dipunken dados guru jogged wonten ing sekolahan.*
 - *Krama Inggil : Bu Niken dipunaturi dados guru beksa wonten ing sekolahan.*

Pencarian

Pencarian data sering juga disebut *tabel look-up* atau *storage and retrieval information* adalah suatu proses untuk mengumpulkan sejumlah informasi yang diperlukan secepat mungkin.

Algoritma pencarian (*searching algorithm*) adalah algoritma yang menerima sebuah argumen kunci dengan langkah-langkah tertentu akan mencari rekaman dengan kunci tersebut. Setelah proses pencarian dilaksanakan, akan diperoleh salah satu dari dua kemungkinan, yaitu data yang dicari ditemukan (*successful*) atau tidak ditemukan (*unsuccessful*).

Metode pencarian data dapat dilakukan dengan dua cara yaitu pencarian internal (*internal searching*) dan pencarian eksternal (*external searching*). Pada pencarian internal, semua rekaman yang diketahui berada dalam penguat komputer sedangkan pada pencarian eksternal, tidak semua rekaman yang diketahui berada dalam penguat komputer, tetapi ada sejumlah rekaman yang tersimpan dalam penyimpanan luar misalnya pita atau cakram magnetis.

Selain itu metode pencarian data juga dapat dikelompokkan menjadi pencarian statis (*static searching*) dan pencarian dinamis (*dynamic searching*). Pada pencarian statis, banyaknya rekaman yang diketahui dianggap tetap, pada pencarian dinamis, banyaknya rekaman yang diketahui bisa berubah-ubah yang disebabkan oleh penambahan atau penghapusan suatu rekaman.

Pencarian Berurutan (Sequential Searching)

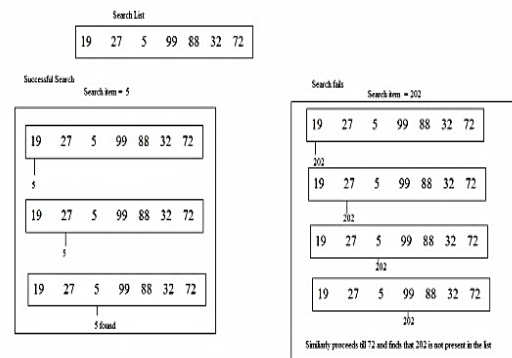
Pencarian berurutan sering disebut pencarian linear merupakan metode pencarian yang paling sederhana. Pencarian berurutan menggunakan prinsip sebagai berikut: data yang ada dibandingkan satu per satu secara berurutan dengan yang dicari sampai data tersebut ditemukan atau tidak ditemukan.

Pada dasarnya, pencarian ini hanya melakukan pengulangan dari 1 sampai dengan jumlah data. Pada setiap pengulangan, dibandingkan data ke-i dengan yang dicari. Apabila sama, berarti data telah ditemukan. Sebaliknya, apabila sampai akhir pengulangan tidak ada data yang sama, berarti data tidak ada. Pada kasus yang paling buruk, untuk N elemen data harus dilakukan pencarian sebanyak N kali pula.

Algoritma pencarian berurutan dapat dituliskan sebagai berikut:

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

Di bawah ini merupakan skema dasar sequential searching



Gambar 1 Skema Dasar Sequential Searching

Proses pencarian dilakukan dengan membandingkan elemen kunci (elemen terbesar yang disimpan di variabel sementara) dengan setiap elemen yang ada di daftar pada sepanjang pencarian. Pencarian berhenti ketika nilai elemen key sama dengan nilai yang dicari.

Beberapa penerapan algoritma *sequential searching* antara lain *sequential searching* tanpa boolean, *sequential searching* dengan boolean, *sequential searching* pada tabel terurut dan *sequential searching* dengan sentinel.

```

Procedure SEQSearchX1 ( Input: T: TabInt, Input N : integer,
                        Output: IX : integer, Input X : integer )
{Mencari harga X dalam Tabel T [1..N] secara sekuensial mulai dari
T1. Hasilnya adalah indeks IX dimana TIX = X (i terkecil) }
{ IX = 0 jika tidak ketemu.}
Kamus :
  i : integer [1..Nmax] (indeks untuk pencarian )
Algoritma :
  i ← 1
  while (i < N) and (Ti ≠ X) do
    i ← i + 1
  { i = N or Ti = X }
  if (Ti = X) then
    IX ← i
  else { Ti ≠ X }
    IX ← 0
    TIDAK KETEMU!

```

Gambar 2 Algoritma sequential searching tanpa Boolean

Pada gambar 2 Elemen tabel yang terakhir diperiksa secara khusus. Jika nilainya sama dengan X, maka IX akan diisi indeks elemen tersebut. Jika tidak sama sama, IX akan diisi nol. Jika IX tidak sama dengan nol, berarti nilai X ditemukan. Proses pada badan pengulangan banyak untuk “maju”/meneruskan indeks berikut. Pencarian dihentikan karena ketemu, atau karena sudah tidak dapat maju. Yang menentukan apakah pencarian ketemu atau tidak adalah kesamaan nilai elemen tabel ketika pencarian dihentikan.

```

Procedure SEQSearchK2 (Input: T: TabInt, N : integer, X : integer
Output: IX : integer, Found : boolean )
{Mencari harga X dalam Tabel T [1..N] secara sekuensial mulai dari
T1, Hasilnya adalah indeks IX dimana TIX =X (i terkecil),
IX = 0 jika tidak ketemu dan sebuah boolean Found (true jika
ketemu).}
Kamus :
i : integer [1..N+1] {indeks untuk pencarian }
Found : boolean {Hasil pencarian }
Algoritma :
Found ← false { awal pencarian, belum ketemu }
i ← 1
while (i ≤ N) and (not Found) do
  if (Ti = X) then
    Found ← true
  else
    i ← i + 1
  { i > N or Found }
  if (Found) then
    IX ← i
  else
    IX ← 0

```

Gambar 3 Algoritma *sequential searching* dengan Boolean

Pada gambar 3 Pengujian elemen array dengan X, jika nilai sama, nilai variabel Found diisi true, ini berarti nilai X ditemukan. Jika tidak, maju/meneruskan indeks berikut. Jika ditemukan, IX akan diisi indeks elemen tersebut, jika tidak IX akan diisi nol. Pencarian dihentikan karena ketemu, atau karena sudah tidak “maju”.

```

Program SEQSearchSorted (Input: T: TabInt, N : integer,
X : integer, Output: IX : integer)
{Mencari harga X dalam Tabel T [1..N] secara sekuensial mulai dr T1
{ Hasilnya adalah indeks IX dimana TIX =X IX = 0 jika tidak ketemu }
Kamus :
i : integer [1..Nmax] {indeks untuk pencarian }
Algoritma :
i ← 1
while (i < N) and (Ti < X)
  i ← i + 1
  { i = N or Ti ≥ X }
  if (Ti = X) then
    IX ← i
  else { Ti * X → Ti > X }
    IX ← 0

```

Gambar 4 Algoritma *sequential searching* tabel terurut

Pada gambar 4 Mencari nilai array yang sudah terurut, Karena terurut maka pencarian dihentikan jika nilai elemen lebih besar dari nilai yang dicari.

Contoh :

1. N=8, T berisi : { 1,3,5,8,12,90,311, 500 }
X=5
Pemeriksaan dilakukan terhadap {1,3,5}
Output : IX = 3
2. N=6, T berisi: {11,30,50,83,99,123} X=70
Pemeriksaan dilakukan terhadap {11,30,50,83,99,123}
Output : IX=0

```

Procedure SEQSearchWithSentinel (Input: T: TabInt, N:integer,X:integer
Output: IX : integer)
{ Mencari harga X dalam Tabel T [1..N] secara sekuensial mulai dari
T1
{ Hasilnya adalah indeks IX dimana TIX =X (IX terkecil) }
{ IX = 0 jika tidak ketemu. Sentinel diletakkan di TN+1 }
Kamus :
i : integer [1..N+1] {indeks untuk pencarian }
Algoritma :
T(N+1) ← X {pasang sentinel }
i ← 1
while (Ti ≠ X) do
  {tidak perlu test terhadap batas i, karena pasti berhenti}
  i ← i + 1
  { Ti = X; harus diperiksa apakah ketemu di sentinel }
  if ( i < N+1) then
    IX ← i {ketemu pada elemen tabel }
  else { i = N+1 }
    IX ← 0 {sentinel, berarti tidak ketemu }

```

Gambar 5 Algoritma *sequential searching* dengan sentinel

Pada gambar 5 *sequential searching* dengan sentinel. Sentinel suatu elemen aktif yang akan menghentikan pengulangan. Elemen fiktif ini harganya adalah sama dengan elemen yang dicari. Posisi sentinel pada tabel, diletakkan di depan jika pencarian mundur atau diletakkan di akhir jika pencarian maju. Teknik Sentinel sangat efisien, terutama jika pencarian dilakukan sebelum penyisipan sebuah elemen yang belum terdapat di dalam tabel.

Jika nilainya ditemukan, diperiksa lagi apakah posisi ketemu di antara elemen tabel yang sebenarnya berarti nilai yang dicari ditemukan. Apabila ditemukan pada elemen terakhir berarti tidak ketemu, karena elemen fiktif/sentinel.

Contoh :

- N=8, T berisi : {1,3,5,8,12,90,3,5} X=5

T dijadikan {1,3,5,8,12,90,3,5,5}

Pemeriksaan dilakukan terhadap {1,3,5}

Output : IX=3

Pengertian Android

Android merupakan subset perangkat lunak untuk perangkat mobile yang meliputi sistem operasi middleware dan aplikasi inti yang di release oleh Google. Sedangkan Android SDK (*Software Development Kit*) menyediakan Tools dan API yang diperlukan untuk mengembangkan aplikasi pada platform Android dengan menggunakan bahasa pemrograman Java.

Dikembangkan bersama antara Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, NVIDIA yang tergabung dalam OHA (*Open Handset Alliance*) dengan tujuan membuat standar terbuka untuk perangkat bergerak (mobile device).

Versi Android

1. Android Versi 1.1

Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam alarm, voice search (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan email. Versi 1.1.

2. Android Versi 1.5 (Cup Cake)

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (Software Development Kit) dengan versi 1.5 (Cupcake). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke headset Bluetooth, animasi layar, dan keyboard pada layar yang dapat disesuaikan dengan sistem.

3. Android Versi 1.6 (Donut)

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang dintegrasikan; CDMA / EVDO, 802.1x, VPN, Gestures, dan Text-to-speech engine, kemampuan dial kontak, teknologi text to change speech (tidak tersedia pada semua ponsel), pengadaan resolusi VWGA.

4. Android Versi 2.0/2.1 (Éclair)

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (Eclair), perubahan yang dilakukan adalah pengoptimalan hardware, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan flash untuk kamera 3,2 MP, digital Zoom, dan Bluetooth 2.1.

Untuk bergerak cepat dalam persaingan perangkat generasi berikut, Google melakukan investasi dengan mengadakan kompetisi aplikasi mobileterbaik (killer apps - aplikasi unggulan). Kompetisi ini berhadiah \$25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik.

5. Android Versi 2.0/2.1 (Éclair)

Pada 20 Mei 2010, Android versi 2.2 (Froyo) diluncurkan. Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi V8 JavaScript engine yang dipakai Google Chrome yang mempercepat kemampuan rendering pada browser, emasangan aplikasi dalam SD Card, kemampuan WiFi Hotspot portabel, dan kemampuan auto update dalam aplikasi Android Market.

6. Android Versi 2.3 (Gingerbread)

Pada 6 Desember 2010, Android versi 2.3 (Gingerbread) diluncurkan. Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (gaming), peningkatan fungsi copy paste, layar antar muka (User Interface) didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (reverb, equalization, headphone virtualization, dan bass boost), dukungan kemampuan Near Field Communication (NFC), dan dukungan jumlah kamera yang lebih dari satu.

7. Android Versi 3.0 (Honeycomb)

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. User Interface pada Honeycomb juga berbeda karena sudah didesain untuk tablet. Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (hardware) untuk grafis. Tablet pertama yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom.

8. Android Versi 4.0 (Ice CreamZ Sandwich)

Diumumkan pada tanggal 19 Oktober 2011, membawa fitur Honeycomb untuk smartphone dan menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan fotografi, mencari emailsecara offline, dan berbagi informasi dengan menggunakan NFC.

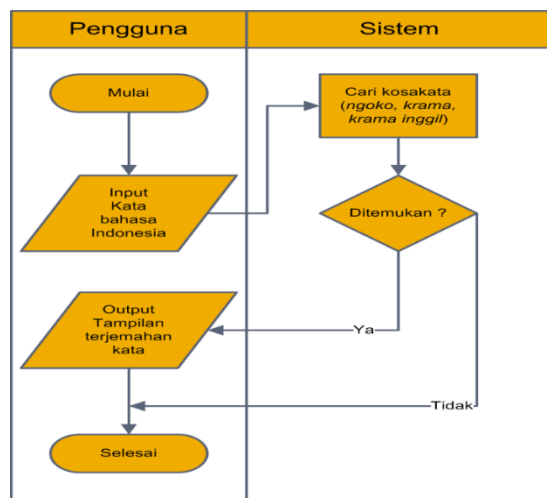
METODOLOGI PENELITIAN

Analisa Sistem

Analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasidan mengevaluasi permasalahan-permasalahan yang terjadi dan kebutuhan kebutuhan yang diharapkan dapat sesuai dengan kebutuhan.

Perancangan aplikasi kamus bahasa Indonesia-Jawa dengan metode *sequential searching* digunakan dalam melakukan pencarian kosakata dari bahasa Indonesia ke bahasa Jawa. Dalam kosakata bahasa Jawa terdapat 3 tingkatan tutur bahasa yang meliputi *ngoko*, *krama madya* dan *krama inggil*. Pencarian kosakata didalam aplikasi kamus bahasa Indonesia-Jawa dilakukan dengan memasukkan kosakata dalam bahasa Indonesia, kemudian dilakukan pencarian yang akan menampilkan kosakata dalam bahasa Jawa yang meliputi tingkat tutur bahasa *ngoko*, *krama madya* dan *krama inggil*. Berikut pada gambar 6

merupakan bisnis proses pada aplikasi kamus bahasa Jawa-Indonesia.



Gambar 6 Bisnis proses aplikasi kamus bahasa Indonesia-Jawa

Perancangan Sistem

Perancangan yang digunakan untuk menggambarkan proses pencarian kosakata didalam aplikasi kamus bahasa Indonesia Jawa menggunakan diagram alir (*flowchart*). Pemodelan sistem pada aplikasi kamus bahasa Indonesia Jawa menggunakan menggunakan UML (*Unified Modeling Language*). Software yang digunakan adalah Software Ideas Modeler.

Adapun struktur perancangan sistem aplikasi kamus bahasa Indonesia Jawa berbasis Android terdiri dari:

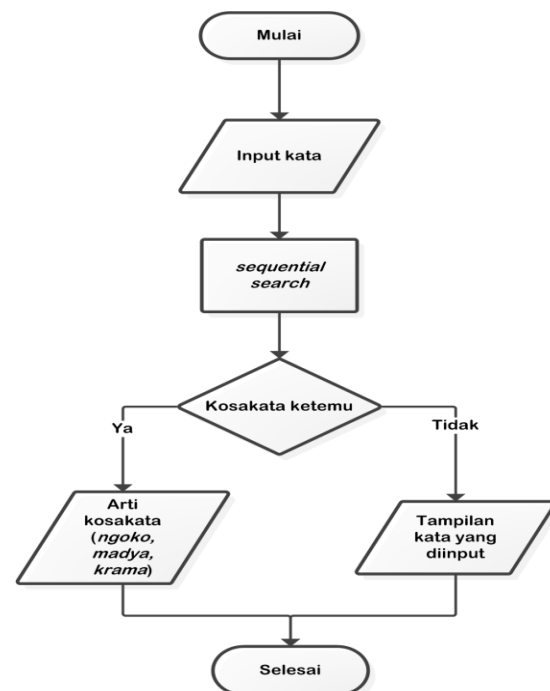
- Flowchart*, digunakan untuk menunjukkan aliran didalam program atau prosedur sistem secara logika.
- Use Case Diagram*, digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case diagram* terdiri atas diagram untuk *use case* dan *actor*. *Actor* merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi
- Activity Diagram*, digunakan untuk memodelkan perilaku *use case* dan *objects* didalam sistem aplikasi kamus bahasa indonesia-jawa yang berbasis Android.
- Sequence Diagram*, menjelaskan secara detail urutan proses yang dilakukan dalam sistem aplikasi kamus bahasa Indonesia-Jawa yang berbasis Android ini untuk mencapai tujuan dari *use case* interaksi yang terjadi antar *class*, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.
- Class Diagram*, merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. *Class diagram* menunjukkan hubungan antar

class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.

- Deployment Diagram*, memiliki fungsi untuk mendeskripsikan arsitektur fisik baik *hardware* maupun *software* dalam sistem serta menggambarkan konfigurasi komponen *software run-time*, peralatan *hardware* yang membentuk sistem.

Flowchart

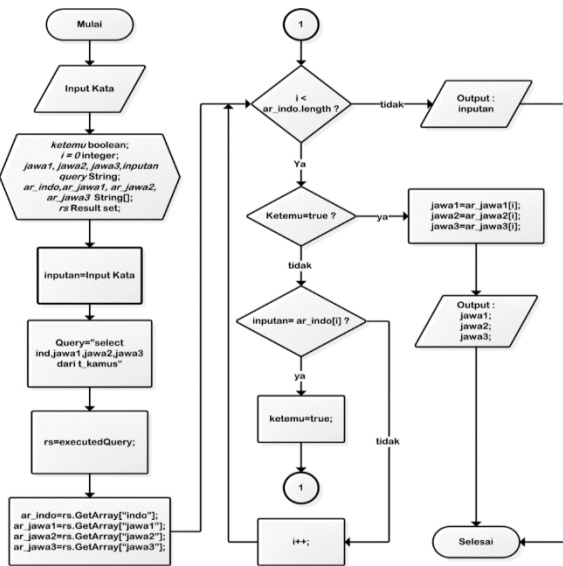
- Flowchart* proses pencarian kata pada aplikasi kamus Indonesia-Jawa



Gambar 7 *Flowchart* proses pencarian kosakata pada aplikasi kamus Indonesia-Jawa

Flowchart pada gambar 7 menunjukkan kata yang di input adalah kata yang dicari, kemudian sistem akan mengerjakan pencarian kata dengan metode *sequential searching*. Apabila kata tersebut ditemukan, maka sistem akan menampilkan arti dari kosa kata yang dicari. Dan jika aplikasi tidak menemukan arti kata yang dicari, maka akan ditampilkan kembali kata yang di input.

2) **Flowchart proses sequential searching**



Gambar 8 Flowchart pencarian kosakata dengan sequential searching

Flowchart pada gambar 8 merupakan flowchart proses pencarian kosakata dengan menggunakan sequential searching, penjelasan mengenai langkah-langkah proses yang dilakukan adalah sebagai berikut :

Setelah mendapat inputan kosakata yang akan dicari, maka dilakukan deklarasi variable yang meliputi jawa1, jawa2, jawa3 (ngoko, karma, krama inggil), ar_indo[], ar_jawa1[], ar_jawa2[], ar_jawa3[], query, rs. Kemudian dilakukan query ke database dengan hasil yang disimpan kedalam result set. Kemudian proses selanjutnya memindahkan hasil dari result set ke variable array yang meliputi ar_indo, ar_jawa1, ar_jawa2, dan ar_jawa3. Setelah itu dilakukan proses pencarian dengan perulangan sebanyak nilai dari data yang terdapat pada array ar_indo. Dilanjutkan dengan memeriksa apakah nilai variabel ketemu adalah true, jika benar maka pencarian dihentikan dilanjutkan dengan proses memindahkan nilai array ar_jawa1, ar_jawa2, ar_jawa3 ke dalam variable jawa1, jawa2, jawa3 sebagai hasil dari pencarian kosakata dan kemudian dikeluarkan nilai dari variable tersebut. Apabila variabel ketemu bernilai false maka dilanjutkan dengan proses membandingkan nilai dari input kata dengan nilai dari array ar_indo dengan index dari variabel i. Jika hasil perbandingan adalah sama maka nilai variabel ketemu adalah true jika tidak maka dilanjutkan perulangan dengan menambahkan nilai i.

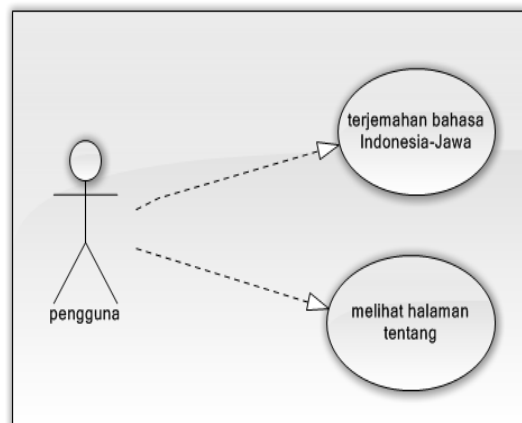
Algoritma pencarian kosakata dengan sequential searching dapat ditulis sebagai berikut:

Tabel 1 Algoritma pencarian dengan sequential searching

1.	Mulai
2.	Input kata dari pengguna sistem
3.	Deklarasi variabel <i>i</i> integer=0, <i>inputan</i> , <i>jawa1</i> , <i>jawa2</i> , <i>jawa3</i> , <i>query</i> String, <i>ar_indo</i> , <i>ar_jawa1</i> , <i>ar_jawa2</i> , <i>ar_jawa3</i> String[]
4.	<i>inputan</i> =Input Kata
5.	<i>Query</i> ="Select ind, jawa1, jawa2, jawa3 from t_kamus"
6.	<i>rs</i> =executed.Query
7.	<i>ar_indo</i> =rs.getArray("indo") <i>ar_jawa1</i> =rs.getArray("jawa1") <i>ar_jawa2</i> =rs.getArray("jawa2") <i>ar_jawa3</i> =rs.getArray("jawa3")
8.	Mengambil nilai result set <i>rs</i> ke dalam variabel <i>ar_indo</i> dari kolom indo, <i>ar_jawa1</i> dari kolom jawa1, <i>ar_jawa2</i> dari kolom jawa2, <i>ar_jawa3</i> dari kolom jawa3
9.	Apakah <i>i</i> < <i>ar_indo.length</i> ? jika tidak ke langkah 11
10.	Apakah <i>ketemu</i> = true ? jika ya ke langkah 12
11.	Apakah input kata equals <i>ar_indo</i> [<i>i</i>] ? jika tidak ke langkah 10
12.	<i>ketemu</i> =true kemudian ke langkah 6
13.	<i>i</i> = <i>i</i> +1 kemudian ke langkah 6
14.	tampilkan output kata awal, kemudian ke langkah 14
15.	variabel <i>jawa1</i> = <i>ar_jawa1</i> [<i>i</i>], <i>jawa2</i> = <i>ar_jawa2</i> [<i>i</i>], <i>jawa3</i> = <i>ar_jawa3</i> [<i>i</i>]
16.	tampilkan <i>jawa1</i> , <i>jawa2</i> , <i>jawa3</i> sebagai hasil pencarian kosakata
17.	Selesai

UML

1) Use Case Diagram



Gambar 9 Use Case Diagram

Diagram use case merupakan pemodelan untuk proses sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat serta digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi.

Aktor pengguna memberikan arah panah ke beberapa use case, antara lain: Actor pengguna dapat memilih menu terjemahan bahasa ke dalam bentuk bahasa Indonesia ke bahasa Jawa, serta Actor pengguna bisa juga memilih menu Tentang yang berisikan informasi tentang aplikasi kamus.

1. Identifikasi Use Case

Identifikasi Use case adalah Use Case yang bekerja pada sistem aplikasi sebagai berikut :

Tabel 2 Deskripsi Use Case

No	Use Case	Deskripsi
1	Terjemahan Indonesia – Jawa	Fungsionalitas untuk terjemahan bahasa indonesia-bahasa jawa
2	Tentang	Fungsionalitas untuk melihat informasi mengenai aplikasi

2. Identifikasi Aktor

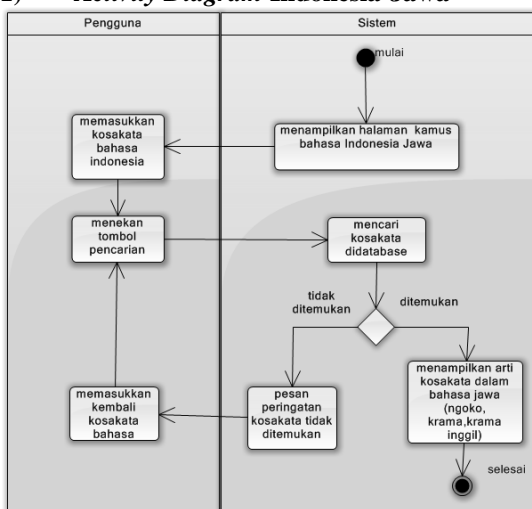
Aktor yang berperan dalam menjalankan aplikasi dapat dilihat pada tabel 3 berikut ini:

Tabel 3 Deskripsi Aktor Pada Sistem

No	Aktor	Deskripsi
1	Pengguna	Merupakan aktor yang menggunakan aplikasi kamus bahasa indonesia-jawa

Activity diagram

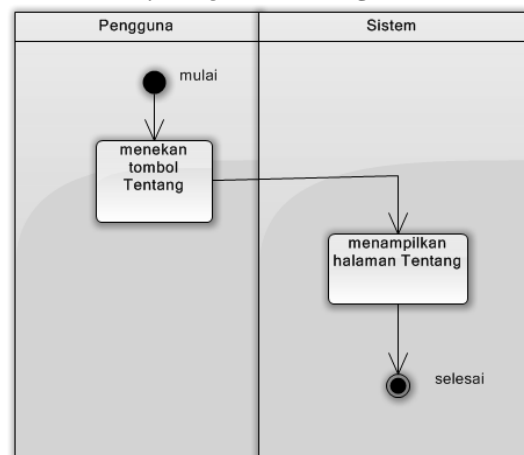
1) Activity Diagram Indonesia-Jawa



Gambar 10 Activity Diagram Bahasa Indonesia-Jawa

Diagram activity bahasa Indonesia Jawa menggambarkan proses yang terjadi pada use case memilih terjemahan bahasa Indonesia ke bahasa Jawa. Awal proses dimulai dengan tampilan sebuah inputan aktif untuk memasukkan kosakata yang akan dicari, pada aplikasi ini jika pengguna tidak menemukan kata yang dicari maka akan muncul pesan peringatan, pengguna harus memasukkan lagi inputan kata yang baru dan jika pencarian telah menemukan kata yang dimaksud maka akan ditampilkan pada inputan pasif, maka aktifitas pencarian selesai dan setelah memilih kosakata yang dimaksud akan keluar arti kata yang telah dipilih.

2) Activity Diagram Tentang

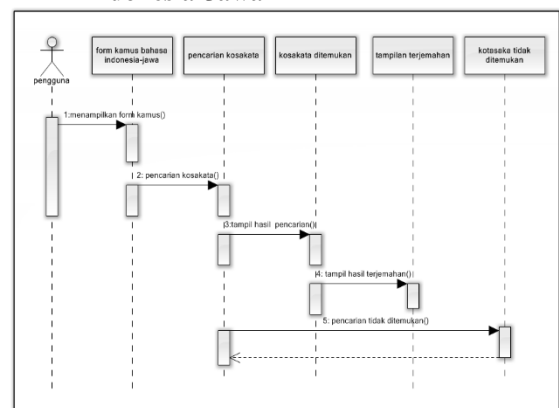


Gambar 11 Activity Diagram Tentang

Pada activity diagram about, pengguna memilih atau menekan tombol about, kemudian sistem akan menampilkan halaman tentang.

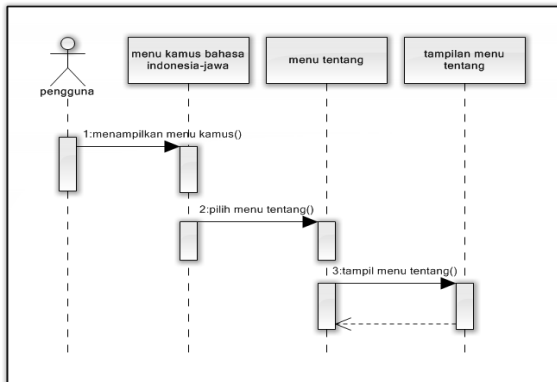
Sequence Diagram

1) Sequence Diagram Terjemahan Bahasa Indonesia-Jawa



Gambar 12 Sequence Diagram Terjemahan Bahasa Indonesia-Jawa

2) **Sequence Diagram ‘Tentang’**



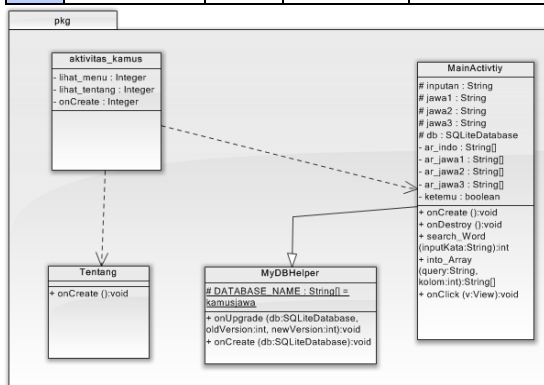
Gambar 13 Sequence Diagram Tentang

Sequence diagram menunjukkan aliran fungsionalitas dalam use case. Proses dalam use case diawali ketika pengguna mencari arti dari sebuah kosakata pada aplikasi kamus bahasa Indonesia-Jawa yang terdiri dari menu menginput pencarian kosakata dan menu about.

Class Diagram

Diagram kelas merupakan diagram struktural yang memodelkan sekumpulan kelas, interface, kolaborasi dan relasinya. Diagram kelas digambarkan dengan kotak, yang pada dasarnya terbagi atas tiga bagian yaitu, Nama Kelas, Atribut, dan Operasi.

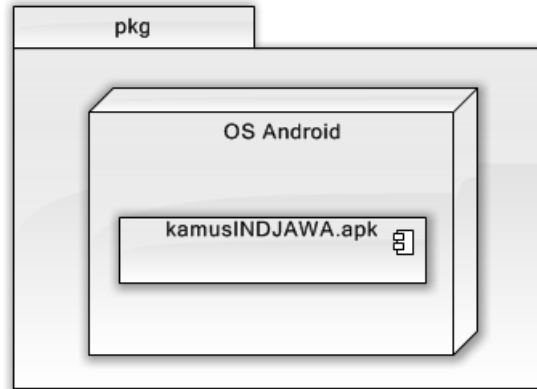
No	Field	Type	Size	
1	No	Int	auto increment	Primary Key
2	Ind	Text	-	
3	Jawa1	Text	-	
4	Jawa2	Text	-	
5	Jawa3	Text	-	
6	Keterangan	Text	-	



Gambar 14 Class Diagram

Deployment Diagram

Diagram Deployment menggambarkan rinci bagaimana komponen di-deploy dalam infrastruktur sistem, dimana komponen akan terletak (pada perangkat keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi Server dan hal-hal yang lain bersifat fisik



Gambar 15 Deployment Diagram

kamusINDJAWA.apk merupakan komponen yang akan didownload kedalam sistem operasi pada perangkat Android yang didalamnya merupakan hasil kompilasi dari Android Developer Tools.

Perancangan Database

Aplikasi database yang digunakan dalam pembuatan aplikasi ini adalah SQLite. Berikut nama tabel yang digunakan beserta field yang terdapat pada tabel.

Tabel 4 tabel t_kamus

HASIL DAN PEMBAHASAN

Kebutuhan Hardware dan Software

Dalam membuat aplikasi kamus bahasa Indonesia-Jawa berbasis sequential searching penulis membutuhkan hardware dan software untuk mendukung dalam kinerja dan proses sistem yang akan dibuat agar kinerja sistem bisa maksimal.

1) **Kebutuhan Perangkat Keras (Hardware)**

Perangkat keras komputer atau hardware sangat mendukung dalam kinerja sistem. Semakin baik spesifikasi perangkat keras yang ada maka kinerja sistem pun akan semakin baik. Spesifikasi perangkat keras yang digunakan adalah sebagai berikut :

1. Handphone dengan sistem operasi Android dengan spesifikasi :

Tabel 5 Spesifikasi perangkat *handphone* yang digunakan

Merk	Samsung
Tipe	Galaxy S 2
Jaringan	GSM 850 / 900 / 1800 / 1900 HSDPA 850 / 900 / 1900 / 2100
Warna	Super AMOLED Plus capacitive touchscreen, 16M colors
Layar	480 x 800 pixels, 4.3 inches (~217 ppi pixel density)
Ukuran/Berat	125.3 x 66.1 x 8.5 mm/116 g
Memori	Internal 16GB storage, 1 GB RAM Eksternal : microSD, up to 32GB
OS	Android OS, v2.3.4 (Gingerbread), v4.0.4 (Ice Cream Sandwich), upgradable to v4.1.2 (Jelly Bean)
CPU	Dual-core 1.2 GHz Cortex-A9, PowerVR SGX540
Kamera	8 MP, 3264x2448 pixels, autofocus, LED flash

2. Memory card
3. Laptop Asus dengan procesor 1.3-GHz Intel Core 2 Duo SU7300, RAM 2GB, Hardisk 320 GB.

2) Kebutuhan Perangkat Lunak (*Software*)

Aplikasi kamus bahasa Indonesia-Jawa ini memerlukan perangkat lunak (*Software*) agar mampu beroperasi sesuai dengan yang diharapkan. Adapun perangkat lunak (*Software*) yang digunakan, yaitu : Sistem Operasi Android minimal 2.3 (Gingerbread)

Pengujian Aplikasi

Pengujian yang dilakukan terhadap sistem dalam penelitian tugas akhir ini adalah pengujian dalam pencarian data dan pengujian secara fungsional.

1) Pengujian pencarian data

Untuk penjelasan lebih lanjut tentang skenario pengujian aplikasi frontend maka dapat dilihat pada tabel 6 berikut :

Tabel 6 Rencana pengujian

Pengujian Aplikasi		
Item Uji	Detail Uji	Jenis Uji
Pencarian arti kosakata bahasa jawa	<i>Keyword</i>	<i>Pencarian data</i>

Berdasarkan rencana pengujian tersebut, maka dapat dilakukan pengujian yaitu sebagai berikut:

1. Pengujian pencarian kosakata bahasa Jawa

Tabel 7 Pengujian Pencarian kosakata bahasa Jawa

Kasus dan hasil uji (data normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
<i>Keyword</i> : 'kamu'	Menampilkan kosakata bahasa Jawa yang dicari berdasarkan <i>keyword</i>	Menampilkan kosakata bahasa Jawa <i>kowe, sampeyan, penjenengan</i>	(√) Diterima () Ditolak
<i>Keyword</i> : 'tanah'	Menampilkan kosakata basa yang dicari berdasarkan <i>keyword</i>	Menampilkan kosakata bahasa Jawa <i>lemah, siti, siti</i>	(√) Diterima () Ditolak
<i>Keyword</i> : 'pergi'	Menampilkan kosakata basa yang dicari berdasarkan <i>keyword</i>	Menampilkan kosakata bahasa Jawa <i>lunga, kesah, tindak</i>	(√) Diterima () Ditolak
<i>Keyword</i> : 'air'	Menampilkan kosakata basa yang dicari berdasarkan <i>keyword</i>	Menampilkan kosakata bahasa Jawa <i>banyu, toya, toya</i>	(√) Diterima () Ditolak
<i>Keyword</i> : 'merah'	Menampilkan kosakata basa yang dicari berdasarkan <i>keyword</i>	Menampilkan kosakata bahasa Jawa <i>abang, abrit, abrit</i>	(√) Diterima () Ditolak
<i>Keyword</i> : 'malam'	Menampilkan kosakata basa yang dicari	Menampilkan kosakata bahasa Jawa <i>bengi, ndalu, ndalu</i>	(√) Diterima () Ditolak

Kasus dan hasil uji (data salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Keyword tidak dimasukkan	Menampilkan pesan "kosakata belum dimasukkan. Silakan masukkan kosakatanya"	Menampilkan pesan yang diharapkan	(√) Diterima () Ditolak
Keyword 'kamu'	Menampilkan pesan "arti kata tidak ditemukan silakan masukkan kata yang lain"	Menampilkan pesan yang diharapkan	(√) Diterima () Ditolak
Keyword: 'tanah'	Menampilkan pesan "arti kata tidak ditemukan silakan masukkan kata yang lain"	Menampilkan pesan yang diharapkan	(√) Diterima () Ditolak
Keyword: 'pergi'	Menampilkan pesan "arti kata tidak ditemukan silakan masukkan kata yang lain"	Menampilkan pesan yang diharapkan	(√) Diterima () Ditolak



Gambar 16 Pengujian dengan keyword 'kamu'

Pada gambar 16 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan menggunakan keyword 'kamu', dimana menampilkan arti kowe(*ngoko*), *sampeyan* (*krama madya*) dan *panjenengan* (*krama inggil*).



Gambar 17 Pengujian dengan keyword 'tanah'

Pada gambar 17 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan menggunakan keyword tanah, dimana menampilkan arti lemah (*ngoko*), *siti* (*krama madya*) dan *siti* (*krama inggil*).



Gambar 18 Pengujian dengan keyword 'pergi'

Pada gambar 18 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan menggunakan *keyword* pergi, dimana menampilkan arti *lunga (ngoko)*, *kesah (krama madya)* dan *tindak (krama inggil)*.



Gambar 19 Pengujian dengan *keyword* 'air'

Pada gambar 19 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan menggunakan *keyword* air, dimana menampilkan arti *banyu (ngoko)*, *toyā (krama madya)* dan *toyā (krama inggil)*.



Gambar 20 Pengujian dengan *keyword* 'merah'

Pada gambar 20 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan menggunakan *keyword* merah, dimana menampilkan arti *abang (ngoko)*, *abrit (krama madya)* dan *abrit (krama inggil)*.



Gambar 21 Pengujian dengan *keyword* 'malam'

Pada gambar 21 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan menggunakan *keyword* merah, dimana menampilkan arti *bengi (ngoko)*, *ndalu (krama madya)* dan *ndalu (krama inggil)*.



Gambar 22 Pengujian dengan tidak memasukkan *keyword*

Pada gambar 22 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan tidak menggunakan *keyword*, dimana menampilkan pesan "Kosakata belum dimasukkan, silakan masukkan kosakatanya".



Gambar 23 Pengujian dengan *keyword* 'kam u'

Pada gambar 23 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan menggunakan *keyword* 'kam u', dimana menampilkan pesan "Arti kata tidak ditemukan, silakan masukkan kata lain".



Gambar 24 Pengujian dengan keyword 't anah'

Pada gambar 24 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan menggunakan keyword 't anah', dimana menampilkan pesan "Arti kata tidak ditemukan, silakan masukkan kata lain".



Gambar 25 Pengujian dengan keyword 'p e r g i'

Pada gambar 25 merupakan hasil pencarian kosakata kamus bahasa Indonesia-Jawa dengan menggunakan keyword 'p e r g i', dimana menampilkan pesan "Arti kata tidak ditemukan, silakan masukkan kata lain".

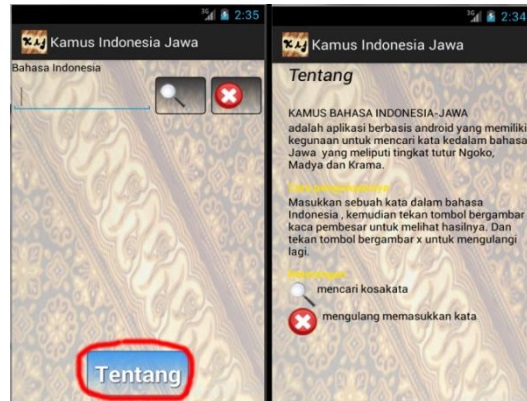
2) **Pengujian Fungsional**

Untuk penjelasan lebih lanjut tentang skenario pengujian fungsional aplikasi frontend maka dapat dilihat pada tabel 5.4 berikut :

Tabel 8 Pengujian Fungsional Aplikasi Kamus Bahasa Indonesia-Jawa

Kasus dan hasil uji

Data masuk an	Yang diharapk an	Pengamat an	Kesimpul an
Pilih menu tentang	Menampil kan informasi tentang aplikasi kamus	Menampil kan informasi tentang aplikasi kamus	(√) Diterima () Ditolak
Tekan tombol pencarian (gambar kaca pembesar)	Melakuka n pencarian kosakata	Menampil kan hasil pencarian	(√) Diterima () Ditolak
Tekan tombol hapus (gambar bulatan bertanda x)	Menghapu s inputan yang telah ada	Inputan kata menjadi kosong	(√) Diterima () Ditolak



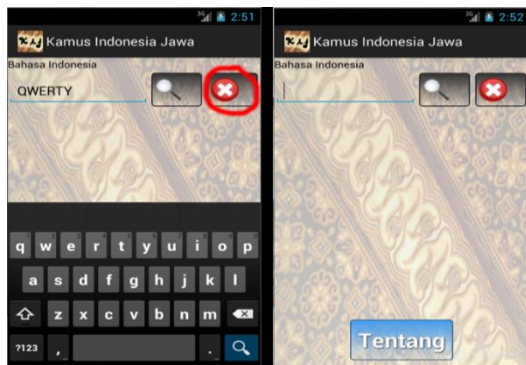
Gambar 26 Pengujian fungsional tombol 'Tentang'

Pada gambar 26 merupakan pengujian fungsi dari tombol 'Tentang', dimana setelah menekan tombol 'Tentang', akan menampilkan halaman 'Tentang' seperti pada gambar 26 bagian kanan.



Gambar 27 Pengujian fungsional tombol cari

Pada gambar 27 merupakan pengujian fungsi dari tombol 'cari', dimana setelah menekan tombol 'cari'; akan menampilkan hasil pencarian arti kosakata seperti pada gambar 27 bagian kanan.



Gambar 28 Pengujian fungsional tombol hapus

Pada gambar 28 merupakan pengujian fungsi dari tombol 'hapus', dimana setelah menekan tombol 'hapus', akan menghilangkan kata yang telah dimasukkan seperti pada gambar 28 bagian kanan.

PENUTUP

Kesimpulan

Berdasarkan uraian pembahasan analisa dan pengujian yang telah dilakukan, maka dapat diambil kesimpulan terhadap aplikasi kamus bahasa Indonesia-Jawa menggunakan *sequential searching* pada *platform* Android yaitu sebagai berikut :

1. Dengan dibangunnya aplikasi ini diharapkan dapat dijadikan media pembelajaran dan sosialisasi sehingga dapat meningkatkan minat masyarakat untuk mempelajari dan lebih memahami bahasa daerah khususnya bahasa Jawa.
2. Aplikasi kamus bahasa Indonesia-Jawa ini dapat dijadikan sebagai media baik sebagai alat bantu penerjemah ataupun untuk pembelajaran bagi setiap individu agar dapat memperluas wawasan terhadap kosakata bahasa Jawa.
3. Dengan dibangunnya aplikasi kamus bahasa Indonesia-Jawa ini dapat memberikan kontribusi untuk mengembangkan ilmu khususnya dalam meningkatkan pengetahuan mengenai arti kosakata bahasa Jawa yang diartikan dari bahasa Indonesia.

Saran

Dalam pembangunan Aplikasi kamus bahasa Indonesia-Jawa ini masih jauh dari sempurna dan masih banyak kekurangan. Oleh

karena itu perlu dilakukan pengembangan dan penyempurnaan lebih lanjut. Adapun saran agar aplikasi ini bisa berfungsi dengan lebih optimal dan lebih menarik sebagai berikut:

1. Menambahkan penulisan arti bahasa Jawanya dalam tulisan Aksara Jawa.
2. Menambahkan fungsi menerjemahkan kosakata dari bahasa Jawa ke dalam bahasa Indonesianya.
3. Pada desain masukkan kata diberikan fungsi *auto suggestion* agar pengguna sistem dapat lebih cepat dalam memasukkan kosakata yang ingin dicari.
4. Penggunaan *relative layout* pada tampilan halaman kamus agar dapat digunakan pada banyak perangkat Android dengan berbagai resolusi layar.

DAFTAR PUSTAKA

- Badudu, J.S. (2007). *Kamus Kata-Kata Serapan Asing dalam Bahasa Indonesia*. Penerbit Buku Kompas. Jakarta.
- H.I, Martina, (2010), *Aplikasi Online Kamus Kedokteran Dengan Menggunakan Metode Binary Search*, Skripsi Teknik Informatika, Universitas Islam Negeri Maliki Malang.
- Haryono Soewardi, (2009), "*Pepak Basa Jawa*". Pustaka Widayatama, Yogyakarta.
- K.S, Okky, (2010), *Aplikasi Pepak Bahasa Jawa Berbasis Android*, Skripsi Teknik Informasi, Politeknik Elektronika Negeri Surabaya Institut Teknologi Sepuluh Nopember, Surabaya.
- Poejosoedarmo, Soepomo dkk. (1979). *Tingkat Tutur Bahasa Jawa*. Jakarta: Pusat
- Sholiq.(2006). *Pemodelan Sistem Informasi Berorientasi Objek Dengan UML*. Graha Ilmu
- Drs. Harjawiyana, Haryana, S.U, (2001) *Kamus Unggah Ungguh Basa Jawa* . Penerbit Kanisius (Anggota IKAPI), Yogyakarta
- Wikipedia, Indonesia, *Kamus*, <http://id.wikipedia.org/wiki/Kamus>, diakses 07 Maret 2013.
- Yusuf, A. (2009), *SejarahKamus*, <http://bukucatatatan-part1.blogspot.com/2009/07/tahu-sejarah-kamus.html>, diakses 08 Maret 2013.

{Halaman sengaja dikosongkan}

{Jurnal **Melek IT**}

(2)

**PERANCANGAN MODEL TATA KELOLA TEKNOLOGI INFORMASI PADA
FAKULTAS TEKNIK UNIVERSITAS WIJAYA KUSUMA SURABAYA BERBASIS
COBIT FRAMEWORK DOMAIN PLANNING AND ORGANIZING****Emmy Wahyuningtyas**

Program Studi Teknik Informatika – Universitas Wijaya Kusuma Surabaya

emmy.cakep@gmail.com**Abstraksi**

Kondisi saat ini Fakultas teknik UWKS telah memiliki 3 (tiga) program studi yaitu Teknik Sipil, Teknologi Industri Pertanian dan Teknik Informatika. Didalam menjalankan kegiatannya fakultas teknik UWKS telah menggunakan IT untuk mendukung beberapa proses antara lain untuk perwalian, pengelolaan nilai, ketatausahaan, pembelajaran *e-learning*.

Guna mendukung hal tersebut dibutuhkan sebuah perencanaan pengembangan teknologi informasi yang cermat dan terintegrasi bagi fakultas teknik UWKS dengan dibuatnya sebuah dokumen rancangan model *IT Governance* untuk sistem Informasi Fakultas teknik UWKS yang terintegrasi dengan menggunakan *COBIT framework* dengan domain *planning and organization*.

Hasil penelitian menunjukkan para responden memilih proses TI PO1 (*Define Strategic IT Plan*). Semua atribut proses TI saat ini berada pada tingkat kematangan 2 serta dihasilkan beberapa strategi perbaikan dan jadwal untuk rencana aksi

Kata Kunci : *e-learning, COBIT framework***1. Pendahuluan**

Kondisi saat ini Fakultas teknik UWKS telah memiliki 3 (tiga) program studi yaitu Teknik Sipil, Teknologi Industri Pertanian dan Teknik Informatika. Didalam menjalankan kegiatannya Fakultas Teknik UWKS telah menggunakan IT untuk mendukung beberapa proses antara lain untuk perwalian, pengelolaan nilai, ketatausahaan, pembelajaran *e-learning*.

Guna mendukung hal tersebut dibutuhkan sebuah perencanaan pengembangan teknologi informasi yang cermat dan terintegrasi bagi fakultas teknik UWKS dengan dibuatnya sebuah dokumen rancangan model *IT Governance* untuk sistem Informasi Fakultas teknik UWKS yang terintegrasi dengan menggunakan *COBIT framework* dengan domain *planning and organization*. Adanya penerapan tata kelola IT ini diharapkan dapat memberikan banyak manfaat bagi fakultas teknik UWKS antara lain mengurangi adanya resiko, menyalurkan IT dengan sasaran fakultas teknik,

membuat operasi fakultas teknik lebih transparan dan meningkatkan efektifitas serta efisiensi.

2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan diatas didapatkan perumusan permasalahan sebagai berikut :

1. Bagaimana mengidentifikasi area potensial yang memiliki prioritas tinggi berkaitan dengan layanan yang dihasilkan.
2. Bagaimana melakukan analisis *maturity level* untuk menemukan gap antara *maturity level* dalam implementasi sistem informasi fakultas teknik dengan *maturity level* yang menjadi tujuan institusi.
3. Bagaimana menyusun rancangan model *IT governance* untuk sistem informasi fakultas teknik UWKS menggunakan *COBIT Framework* dengan domain *planning and organization*.

3. Tinjauan Pustaka

3.1 Definisi Visi dan Misi

Sebuah organisasi harus mempunyai visi dan misi untuk menentukan arah perkembangannya. Definisi dari visi (Indrajit, 2000) merupakan sesuatu yang dicanangkan oleh pendiri organisasi. Namun yang harus diperhatikan, visi bukanlah mimpi, namun sesuatu yang mungkin terwujud. Sedangkan misi ditetapkan sebagai jawaban terhadap visi yang telah ditetapkan sebelumnya. Misi masih merupakan sesuatu yang memiliki arti global dan cenderung generik. Oleh karena itu, ditentukan beberapa objektif yang ingin dicapai dalam berbagai hal sehubungan dengan misi yang dicanangkan tersebut.

3.2 Strategi Bisnis

Strategi didefinisikan sebagai (Cannon, 2006) :
“an adaptation of behavior or structure with an elaborate and systematic plan of action”.

Strategi juga dapat didefinisikan sebagai suatu rangkaian kegiatan yang terintegrasi dan ditujukan untuk meningkatkan faktor-faktor yang menentukan tujuan dan kemampuan perusahaan (Edwards, 1995).

Business Strategy merupakan dokumen yang harus dijadikan landasan berpijak utama dalam pembuatan *I/T Strategy* karena dalam dokumen tersebut disebutkan visi dan misi perusahaan beserta target kinerja masing-masing fungsi pada struktur organisasi (Indrajit, 2000).

3.3 Sistem Informasi Fakultas Teknik

Dalam teori organisasi modern, perusahaan yang ingin berkembang saat ini harus mengubah filosofi cara memandang aktivitas internal perusahaan, dari yang berbasis hierarki (atau fungsional) untuk keperluan manajemen internal, menjadi berbasis proses yang berorientasi pada kepuasan pelanggan. Salah satu strategi perubahan yang mendukung proses tersebut adalah “internal integration”, yaitu jenis strategi perubahan yang tujuan utamanya untuk melakukan integrasi antara fungsi-fungsi atau departemen-departemen yang ada dalam perusahaan (Indrajit, 2000). Selanjutnya (Indrajit, 2000) mengemukakan bahwa tiga hal pokok yang perlu dipahami secara menyeluruh apa yang harus diperhatikan dan dipertimbangkan untuk menghasilkan sebuah sistem informasi yang terintegrasi dengan baik adalah:

Pertama, sistem informasi, merupakan definisi secara jelas dan terperinci sehubungan dengan jenis-jenis informasi apa saja yang

dibutuhkan oleh perusahaan (dalam hal ini fakultas teknik) dan hal-hal yang berkaitan dengan (kecepatan proses pengolahan data menjadi informasi, tingkatan detail informasi, cara penampilan informasi, volume dan transaksi informasi, penanggung jawab informasi, dsb.)

Kedua, teknologi informasi, meliputi komponen-komponen perangkat keras (komputer, infrastruktur, alat komunikasi, dan lain-lain) dan perangkat lunak (aplikasi, sistem operasi, database, dan lain-lain) yang harus tersedia untuk menghasilkan sistem informasi yang telah didefinisikan.

Ketiga, manajemen informasi, menyangkut perangkat manusia (*brainware*) yang akan mengimplementasikan sistem informasi yang dibangun dan mengembangkan teknologi informasi sejalan dengan perkembangan perusahaan (institusi) di masa mendatang.

Secara mendasar (Lancaster, 1988) membagi kegiatan-kegiatan yang ada pada fakultas teknik itu menjadi dua kelompok yaitu kegiatan pertama berhubungan dengan organisasi dan pengawasan sumber-sumber informasi. Kegiatan-kegiatan ini berupa layanan teknis yang menghasilkan berbagai macam alat bantu yang akan membantu kegiatan fakultas teknik. Kelompok keduanya, yaitu layanan publik. Layanan publik kemudian dibagi lagi menjadi dua kelompok: *demand service* dan *notification service*. Layanan yang pertama bersifat pasif menunggu respon atas permintaan para pengguna, sedangkan yang kedua lebih dinamis dengan mencoba mendesain layanan untuk diinformasikan kepada para pengguna sehingga menjadi tertarik.

Untuk itu dalam lingkungan fakultas teknik era globalisasi sekarang ini, perencanaan merupakan hal yang lebih penting dan sangat diperhatikan dari pada yang terjadi sebelumnya. Teknologi telah berkembang dengan sangat cepat untuk memasukan layanan dan sistem yang lebih luas yang membuat fakultas teknik dapat mengorganisir informasi yang diciptakan secara lokal dan akses informasi yang tersebar secara global. Penting sekali untuk mengembangkan perencanaan teknologi dasar sebagai tahap awal dalam mengidentifikasi sistem dan layanan mana yang paling baik guna memenuhi kebutuhan pengguna dan memenuhi misi fakultas teknik tersebut (Cohn, 2001).

3.4 Strategi Teknologi Informasi

Strategi teknologi informasi lebih menekankan pada garis besar bagaimana kebutuhan organisasi atas informasi dan sistem akan disupport teknologi atau esensinya lebih menekankan pada ‘*IT supply*’ (Ward, 2002).

Beberapa alasan dibutuhkannya strategi teknologi informasi adalah (Ward, 2002) :

1. Sistem investasi yang dibuat tidak mendukung tujuan bisnis.
2. Hilangnya kontrol IS/IT.
3. Sistem tidak terintegrasi.
4. Tidak berarti untuk menetapkan prioritas proyek/sumberdaya IS dan perubahan rencana berdasar produktifitas yang rendah.
5. Bukan mekanisme untuk menentukan tingkat optimal dari sumberdaya, tetapi merupakan 'system supply' yang terbaik.
6. Informasi manajemen yang jelek.
7. Kesalahpahaman yang menyebabkan konflik antara user dan spesialis IT.
8. Strategi teknologi yang tidak logis dan ketidakleluasaan pilihan.
9. Membuat investasi infrastruktur tidak mencukupi.
10. Evaluasi seluruh proyek dengan basis keuangan saja.
11. Masalah investasi IS/IT yang mendatangkan konflik antar bagian organisasi.
12. Penyesuaian secara lokal investasi yang memberi keuntungan, tetapi nyatanya secara bisnis keseluruhan tidak produktif.
13. Sistemnya rata-rata, memiliki jangkauan yang pendek dan kebutuhan secara menyeluruh sangat besar di pengeluaran untuk membangun ulang IS/IT.

3.5 Tata Kelola (IT Governance)

Gartner menawarkan sebuah konsep *governance* yang diberi nama "*Gartner's Integrated Planning Suite*". Dalam kerangka ini, ada empat aspek yang saling terkait satu dengan lainnya sehubungan dengan prinsip *governance* yang ingin ditegakkan, dimana masing-masing memiliki relasi keterkaitan sebagai berikut (Indrajit, 2004) :

1. *Strategic Planning*
Rencana strategis sebuah perusahaan akan memicu dan mengarahkan disusunnya sebuah rencana pengembangan teknologi informasi. Dengan berpedoman kepada visi, misi, dan tujuan perusahaan maka akan di dapat gambaran yang jelas mengenai peranan dan teknologi informasi seperti apa yang akan dikembangkan. Detil dari rencana tersebut dapat dijabarkan dalam sebuah dokumen Rencana Induk Pengembangan Teknologi Informasi atau Master Plan IT.
2. *Enterprise Architecture*
Merupakan keseluruhan komponen dan hubungan keterkaitan satu dengan lainnya yang membentuk sebuah sistem teknologi informasi

korporat. Di dalam arsitektur tersebut diperlihatkan pula filosofis pembangunan sistem secara "rumah tumbuh" yang akan dikembangkan oleh perusahaan sesuai dengan kekuatan dan keterbatasan sumber daya yang dimiliki.

3. *Portofolio Performance Management*

Karena begitu banyaknya komponen dalam arsitektur teknologi informasi yang harus dibangun dimana terbagi menjadi sejumlah kategori seperti perangkat lunak, perangkat keras, dan perangkat manusia maka diperlukan suatu pendekatan portofolio agar terjadi optimalisasi proses pengembangan. Konsep tersebut dikembangkan berakar dari keanekaragaman perspektif atau pandangan mengenai *nature* dari teknologi informasi yang ingin dibangun, seperti dilihat dari segi prioritas, fungsi, utilisasi, kebutuhan, demografi, stakeholder, karakteristik sumber daya, aspek perencanaan, dan lain sebagainya.

4. Dalam perkembangannya keputusan yang diambil berdasarkan prinsip manajemen portofolio ini akan diukur kinerjanya, terutama terkait dengan bagaimana keputusan penerapan teknologi informasi tersebut akan berpengaruh terhadap kinerja bisnis perusahaan secara menyeluruh. Sehingga dapat dikatakan bahwa manajemen portofolio tersebut akan mempengaruhi *strategic planning* yang disusun.

3.6 COBIT

ISACA (*Information System Audit and Control Association*) memperkenalkan sebuah kerangka untuk mengelola IT Governance di sebuah perusahaan yang dikenal dengan nama COBIT (*Control Objectives for Information and Related Technologies*) (Indrajit, 2004).

COBIT adalah salah satu methodology yang memberikan kerangka dasar dalam menciptakan sebuah Teknologi Informasi yang sesuai dengan kebutuhan organisasi dengan tetap memperhatikan faktor-faktor lain yang berpengaruh. Pada dasarnya COBIT dikembangkan untuk membantu memenuhi berbagai kebutuhan manajemen terhadap informasi dengan menjembatani kesenjangan antara resiko bisnis, kontrol dan masalah teknik (Swastika, 2007).

COBIT dapat menyediakan seperangkat praktek yang dapat diterima pada umumnya karena dapat membantu para direktur, eksekutif dan manager meningkatkan nilai IT dan mengecilkan resiko. Saat ini COBIT telah mencapai edisi ke-4, COBIT 4.0 ini juga mencakup bimbingan bagi para direktur dan semua level manajemen dan terdiri atas empat seksi (ITGI, 2007) :

1. Gambaran luas mengenai eksekutif
2. Kerangka kerja
3. Isi utama (tujuan pengendalian, petunjuk manajemen dan model kedewasaan)
4. Appendix (pemetaan, ajuan silang dan daftar kata-kata)

Isi utama dibagi lagi menurut proses 34 IT dan memberikan gambaran yang sempurna mengenai cara mengendalikan, mengelola dan mengukur masing-masing proses, selain itu standar COBIT 4.0 juga :

- a. Menganalisa bagaimana tujuan pengendalian dapat dipetakan ke dalam lima wilayah penentuan IT agar dapat mengidentifikasi gap potensial.
- b. Menyesuaikan dan memetakan COBIT ke standar yang lain (ITIL, CMM, COSO, PMBOK, ISF and ISO 17799)
- c. Mengklarifikasikan indikator tujuan utama (KGI) dan indikator hubungan kinerja utama (KPI), dengan mengenal bagaimana KPI dapat bergerak mencapai KGI.
- d. Menghubungkan tujuan bisnis, IT dan proses IT (penelitian mendalam di delapan industri dengan pandangan yang lebih jelas tentang bagaimana proses COBIT mendukung tercapainya tujuan IT spesifik dan dengan perluasan, tujuan bisnis).

4. Metode Penelitian

4.1 Studi Literatur

Studi literatur ini dilakukan dengan mempelajari COBIT dari dokumen-dokumen, buku, diktat, e-book, jurnal dan juga memanfaatkan fasilitas yang ada di internet. Untuk landasan yang berisi data-data yang digunakan dalam penelitian yang berhubungan dengan institusi atau fakultas teknik diperoleh dari arsip-arsip institusi kaitannya dengan muatan visi, misi dan juga rencana pengembangan jangka pendek maupun jangka panjang.

4.2 Kuesioner dan Wawancara

Untuk mengetahui kondisi di fakultas teknik UWKS dilakukan dengan penyebaran kuisisioner dan juga wawancara langsung dengan beberapa pihak yang terkait. Langkah ini untuk mengetahui kondisi sistem berjalan dan juga kondisi sistem yang diharapkan.

Kuisisioner yang dibagikan meliputi 2 aspek antara lain *Management Awareness*, untuk mendapatkan data dan opini dari pihak manajemen institusi dan pimpinan fakultas teknik mengenai penerapan

sistem informasi fakultas teknik dan *IT Control Diagnostics*, untuk mendapatkan data dan opini dari beberapa pihak yaitu pengguna dan pengelola tentang implementasi sistem informasi fakultas teknik

4.3 Analisis Kesenjangan

Analisis ini dilakukan terhadap *maturity level* dari domain yang dianalisis, yaitu domain *Planning and organization*, untuk kemudian merumuskan *Key Goal Indicators* (KGIs), *Key Performance Indicators* (KPIs) dan juga *Critical Succes Factors* (CSFs).

4.4 Merancang Model IT Governance untuk

Sistem Informasi Fakultas teknik

Pada tahapan ini adalah tahapan akhir dari penelitian ini yaitu merancang model tata kelola IT (*IT Governance*) untuk sistem informasi fakultas teknik UWKS dengan mengacu pada standart *framework* COBIT pada domain *Planning and Organization*.

5. Pengolahan Dan Analisa Data

5.1 Pemilihan Prioritas Proses TI

Dalam penelitian ini dilakukan pemilihan prioritas proses TI pada COBIT yang akan dibuat tata kelola. Namun dalam menentukan prioritas maka perlu dilakukan analisa hubungan antar setiap proses TI yang telah ditentukan dalam IT Goals yang adda pada COBIT dengan perancangan tata kelola untuk perencanaan dan organisasi (Plan and Organize) pada Fakultas Teknik Universitas Wijaya Kusuma Surabaya.

5.1.1 Proses TI yang Relevan

Setiap proses TI (PO1, PO4, PO10, ME1 dan ME4) yang telah ditentukan pada IT Goals untuk merespon kebutuhan bisnis organisasi agar sejalan dengan strategi TI (ITGI, 2007). Berikut ini hubungan setiap proses TI dalam COBIT untuk IT Goals yang telah ditentukan :

a. PO1 Define Strategic IT Planning

Dengan dapat mengontrol proses TI dalam menentukan strategi perencanaan TI yang memenuhi persyaratan bisnis di FT UWKS, maka dapat mempertahankan dan memperluas strategi bisnis dan kebutuhan tata kelola dengan mempertimbangkan tentang manfaat, biaya dan resiko.

b. PO4 Define IT Processes, Organisation and Relationship

Proses TI untuk PO4 yaitu menentukan proses TI, organisasi TI serta hubungan TI dengan pengguna maupun pemasok yang sesuai dengan persyaratan bisnis TI dalam menanggapi strategi bisnis yang suatu saat bisa berubah serta dibangun secara transparan, fleksibel dan responsif.

c. PO10 Manage Projects

Pada proses ini manajemen proyek dikelola untuk memastikan bahwa hasil proyek sesuai dengan kerangka waktu, biaya dan kualitasnya. Dengan manajemen proyek maka dapat diketahui seberapa banyak tenaga yang dibutuhkan serta macam keahlian untuk mengelola proyek TI sehingga akan berpengaruh pada struktur organisasi TI.

d. ME1 Monitor and Evaluate IT Performance

Mengawasi serta mengevaluasi secara internal dapat melindungi pencapaian tujuan TI dan mematuhi hukum, aturan dan kontrak yang berlaku untuk proyek TI. Dengan demikian strategi TI akan tetap sesuai jalurnya.

e. ME4 Manage IT Performance

Menyediakan tata kelola TI yang sesuai persyaratan bisnis TI bertujuan untuk mengintegrasikan tata kelola TI dengan tata kelola Fakultas.

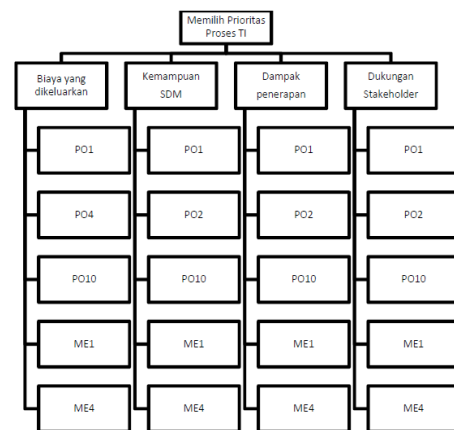
5.1.2 Pemilihan Prioritas Proses TI Menggunakan AHP

Untuk memperkuat keputusan yang telah diambil pihak manajemen maka dilakukan pemilihan prioritas proses TI menggunakan metode AHP (*Analytical Hierarchy Process*). Langkah-langkah penentuan bobot kepentingan dengan metode AHP sebagai berikut:

1. Mengidentifikasi Permasalahan

Hasil langkah pertama dalam menggunakan AHP diperlukan definisi awal sebagai berikut:

Obyektif : menentukan proses TI
 Kriteria : biaya yang dikeluarkan, kemampuan SDM, Dampak Penerapan, Dukungan Stakeholder
 Alternatif : PO1, PO4, PO10, ME1, ME4



Gambar 4.1 Hirarki Pemilihan Proses TI

2. Penyusunan Kuesioner

Terdapat lima kuesioner yang berisi tingkat perbandingan antar kriteria dan perbandingan control objective untuk setiap kriteria. Skala banding yang digunakan berada pada rentang 1 – 9. Pada perhitungan kuesioner yang membandingkan tingkat kepentingan antar kriteria, terdapat pertanyaan perbandingan untuk menentukan bobot kepentingan dari setiap kriteria yang telah diidentifikasi sebelumnya, yaitu Biaya yang dikeluarkan, Kemampuan SDM, Dampak Penerapan serta Dukungan Stakeholder.

3. Penentuan Responden

Responden ditentukan berdasarkan keahlian pengalaman mereka terhadap kriteria yang digunakan, yaitu: Dekan, Pembantu Dekan Bidang Akademik, Pembantu Dekan Bidang Administrasi, Kepala Tata Usaha, Ketua Program Studi Teknik Informatika.

4. Menentukan Matriks Perbandingan Berpasangan

Dengan menggunakan ketentuan tersebut diperoleh matriks rata-rata perbandingan berpasangan sebagai berikut:

a. Matriks perbandingan berpasangan antar kriteria

GOAL	BIAYA	SD	DAMPA	STAKE_
	A	M	K	H
BIAYA	1.00	1.93	5.16	4.83
SDM	0.52	1.00	7.24	5.85
DAMPAK	0.19	0.14	1.00	2.95
STAKE_	0.23	0.17	0.34	1.00
H				

b. Matriks perbandingan antar proses TI untuk kriteria "Biaya yang dikeluarkan"

GOAL	PO1	PO4	PO10	ME1	ME4
PO1	1.00	1.25	5.16	6.12	7.36
PO4	0.13	1.00	3.27	6.77	3.32
PO10	0.03	0.32	1.00	2.95	1.93
ME1	0.03	0.15	0.35	1.00	2.67
ME4	0.14	0.30	0.52	0.37	1.00

c. Matriks perbandingan antar proses TI untuk kriteria "Kemampuan SDM"

GOAL	PO1	PO4	PO10	ME1	ME4
PO1	1.00	2.85	2.36	3.37	6.12
PO4	0.12	1.00	5.35	3.35	4.66
PO10	0.05	0.19	1.00	7.24	1.25
ME1	0.10	0.19	0.35	1.00	7.24
ME4	0.16	0.21	0.52	0.14	1.00

d. Matriks perbandingan antar proses TI untuk kriteria "Dampak Penerapan"

GOAL	PO1	PO4	PO10	ME1	ME4
PO1	1.00	2.85	4.99	6.12	3.50
PO4	0.06	1.00	3.62	8.14	3.16
PO10	0.03	0.28	1.00	3.74	1.93
ME1	0.03	0.12	0.27	1.00	5.35
ME4	0.29	0.32	0.52	0.19	1.00

e. Matriks perbandingan antar proses TI untuk kriteria "Dukungan Stakeholder"

GOAL	PO1	PO4	PO10	ME1	ME4
PO1	1.00	6.12	6.43	2.41	2.22
PO4	0.08	1.00	3.27	3.62	4.08
PO10	0.08	0.31	1.00	5.52	3.94
ME1	0.21	0.28	0.18	1.00	2.85
ME4	0.45	0.25	0.25	0.35	1.00

5. Perhitungan Bobot seluruh kriteria dan alternatif Dalam perhitungan bobot seluruh kriteria dan alternatif digunakan bantuan software QM for Windows ver. 2.0. Hasil perhitungan bobot kriteria dan alternatif secara individual menggunakan QM sebagai berikut :

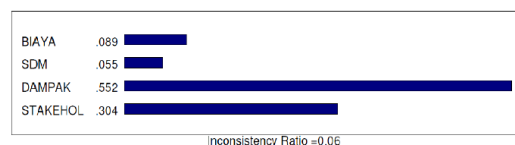
a. Perhitungan bobot untuk kriteria Biaya, SDM, Dampak serta Stakeholder adalah sebagai berikut :

Compare the relative IMPORTANCE with respect to: GOAL

	SDM	DAMPAK	STAKEHOL
BIAYA	2.0	(5.0)	(5.0)
SDM		(7.0)	(6.0)
DAMPAK			3.0

Row element is ... times more than column element unless enclosed in ()

Abbreviation	Definition
Goal	Memilih Prioritas Proses TI
BIAYA	Biaya yang dikeluarkan dalam menerapkan Proses TI
SDM	Kemampuan dan Ketersediaan SDM untuk menerapkan Proses TI
DAMPAK	Pengaruh Penerapan Proses TI terpilih untuk kinerja Tata Kelola
STAKEHOL	Dukungan Manajemen RS.XYZ untuk menerapkan Proses TI



Gambar 4.2 Hasil Perhitungan Bobot untuk seluruh kriteria

Penggunaan QM menunjukkan bahwa rasio konsistensi yang dihasilkan sebesar 0.06 masih di bawah rasio yang diperkenankan oleh Saaty yaitu sebesar 0.1, sehingga besaran bobot yang dihasilkan dapat dipercaya.

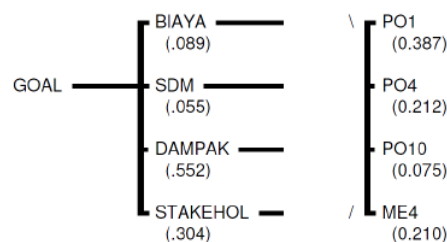
Kriteria Dampak Penerapan memiliki bobot kepentingan paling besar yaitu 0.552, hal ini berarti manajemen yang menjadi responden memiliki persepsi yang sama bahwa kriteria tersebut merupakan parameter yang sangat mempengaruhi penentuan control objectives yang hendak diterapkan.

Sedangkan kriteria untuk Dukungan Stakeholder yang cukup besar dibanding dengan Biaya maupun SDM yaitu 0.304. Hal ini menunjukkan bahwa Dukungan Stakeholder sedikit mempengaruhi penentuan Proses TI.

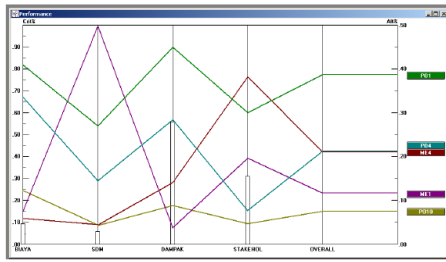
Untuk kriteria SDM dan Biaya memiliki bobot kepentingan yang rendah, yang menunjukkan bahwa hal kedua kriteria ini tidak menjadi kendala dalam menentukan prioritas.

6. Penentuan prioritas Control Objective

Penentuan prioritas proses TI secara individual pada masing-masing kriteria memberikan hasil yang berbeda. Proses TI dengan prioritas tertinggi pada kriteria Biaya dan Dampak Penerapan adalah PO1 (mendefinisikan strategi TI).



Gambar 4.7 Pohon Prioritas Control Objective



Gambar 4.8 Diagram Tingkat Prioritas Proses TI

C1-T	C2	C3	C4	C5	C6	C7	C8
Kode	AC-IS	PSP-IS	TA-IS	SE-IS	RA-IS	GSM-IS	Jumlah
1 DVE	3	2	1	1	2	1	10
2 DPL	3	3	2	3	3	3	17
3 DSD	2	3	2	2	3	3	15
4 DUA	2	1	1	2	2	2	10
5 DUT	3	4	4	3	3	3	20
6 MEA	2	2	1	2	2	1	10
7 MEAK	1	1	3	3	2	2	12
8 MEH	1	1	2	1	2	3	10
9 MEL	1	1	3	3	2	2	12
10 MEP	1	1	2	1	2	3	10
11 MEPS	3	2	2	2	2	1	12
12 MES	2	3	2	2	3	3	15
13 MIB	3	2	3	3	3	3	17
14 MIF	2	2	2	2	1	1	10
15 MII	2	2	1	1	2	2	10
16 MIJ	3	3	3	3	3	3	18
17 MIP	1	1	2	1	3	2	10

Gambar 4.9 Rekap data kondisi saat ini (*as-is*) dalam Minitab

5.2 Analisa Tingkat Kematangan (Maturity Level)

5.2.1 Kuesioner Analisa Tingkat Kematangan Pengembangan obyek pertanyaan dan pilihan jawaban kuesioner dilakukan dengan langkah-langkah sebagai berikut :

1. Mendeskripsikan pernyataan tingkat kematangan proses PO1
2. Selanjutnya dilakukan pengembangan sehingga didapatkan definisi pernyataan kematangan yang merepresentasikan semua atribut pada semua tingkat kematangan.
3. Selanjutnya disusun matriks atribut kematangan
4. Mentranslasikan matriks kedalam bentuk pertanyaan dan jawaban yang benar

No	Status	Alpha	Reliabilitas
1	As-Is	0.7888	Reliabel
2	To-Be	0.7549	Reliabel

Tabel 4.2 Hasil Perhitungan *Cronbach Alpha*

5.2.2 Pelaksanaan Kuesioner

Distribusi kuesioner dilakukan dengan mengacu pada tabel RACI sesuai dengan panduan COBIT, yang selanjutnya dipetakan ke dalam peran dalam organisasi.

5.2.3 Hasil Uji Reliabilitas dan Validitas

Diuji menggunakan nilai Cronbach Alpha. Data yang dianalisa harus memiliki nilai alpha > 0.6 agar data tersebut dianggap reliabel. Proses uji reliabilitas dimulai dengan merekapitulasi jawaban responden ke dalam worksheet Minitab dengan hasil sebagai berikut :

Dengan demikian dapat disimpulkan bahwa data yang dianalisa dapat memberikan keyakinan atas konsistensi dalam mengukur gejala yang sama.

Uji validitas dilakukan dengan uji korelasi Pearson, yaitu dengan mengkorelasikan skor butir pada kuesioner dengan skor totalnya. Untuk semua jawaban pada atribut dihitung nilai r (korelasi dengan skor total) dan dibandingkan dengan nilai r pada tabel.

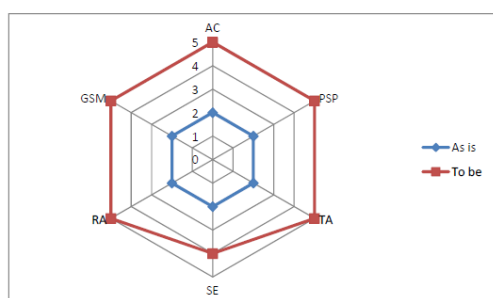
Pearson product moment Correlation Coefficient untuk jumlah responden yang sesuai. Jika nilai r-hitung > r-tabel, maka item dikatakan valid dengan taraf signifikansi sebesar 5%.

5.2.4 Penghitungan Tingkat Kematangan

Pada tahap ini dilakukan penghitungan nilai kematangan dengan mencari nilai rerata jawaban pada masing-masing atribut. Selanjutnya dilakukan pembobotan terhadap hasil rerata tersebut sesuai dengan Nilai Kematangan.

No	Atribut	Nilai Kematangan		Tingkat Kematangan	
		As-is	To-be	As-is	To-be
1	AC	2.06	4.71	2	5
2	PSP	2	4.64	2	5
3	TA	2.12	5	2	5
4	SE	2.06	4.36	2	5
5	RA	2.35	4.71	2	5
6	GSM	2.24	4.64	2	5
	Rerata	2.14	4.68	2	5

Tabel 4.3 Nilai Kematangan PO1



Gambar 4. 10 Representasi Tingkat Kematangan

5.2.5 Analisa Kondisi Saat Ini

Secara umum kondisi kematangan saat ini pada semua atribut PO1 berada pada level 2, yaitu *Repeatable but Intuitive*. Sesuai hasil observasi, kondisi saat ini pada proses PO1 dapat disimpulkan sebagai berikut :

1. Perencanaan strategis telah dikomunikasikan oleh manajemen dalam bentuk formal (AC)
2. Kebijakan standarisasi atau prosedur yang berkaitan dengan perencanaan strategis TI, mulai direncanakan dan disusun (PSP)
3. Alat bantu yang berkaitan mulai direncanakan dan disusun (TA)
4. Kebutuhan ketrampilan minimal telah diidentifikasi atas dasar respon dari kebutuhan manajemen (SE)
5. Peran dan tanggung jawab terkait perencanaan strategis TI disusun atas dasar respon dari manajemen, jadi pelaksanaannya masih bersifat informal. (RA)

5.2.6 Analisa Kondisi Yang Diharapkan

Dari hasil kuesioner dapat disimpulkan bahwa manajemen menginginkan hampir semua atribut pada tingkat kematangan 5-Optimised. Kecuali untuk atribut Skill and Expertise berada pada tingkat 4-Managed and Measurable.

Kondisi yang diharapkan untuk PO1 berada di tingkat 5, yaitu :

1. Perencanaan strategis selalu dipikirkan, dikomunikasikan dan dikembangkan secara periodik dan berkelanjutan serta memiliki aturan yang jelas, yakni mendukung strategi bisnis institusi serta menghasilkan nilai bisnis yang tak terlihat melalui investasi TI (AC)
2. Kebijakan, standarisasi dan prosedur yang berkaitan dengan perencanaan strategis TI telah dipikirkan, dikomunikasikan, dipertimbangkan dan disusun secara periodik dan berkelanjutan. Aturan-aturan yang telah ditetapkan dan menjadi standar dalam berhubungan dengan berbagai aktifitas yang bersifat eksternal (PSP)

3. Alat bantu yang berkaitan dengan perencanaan strategis TI telah dipikirkan, dikomunikasikan, dipertimbangkan dan disusun secara periodik dan berkelanjutan dan dipasang agar berfungsi secara otomatis (TA)
4. Peran dan tanggung jawab terkait perencanaan strategis TI dirancang sedemikian rupa, dikomunikasikan, dipertimbangkan dan diperbarui secara periodik atau berkelanjutan (RA)

5.3 Analisa Kesenjangan (Gap Analysis)

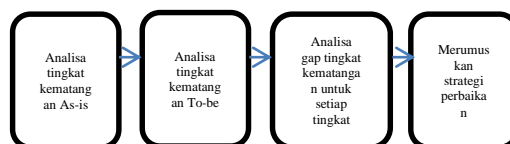
Mengacu pada nilai-nilai kematangan yang telah diperoleh, rekomendasi tindakan dikelompokkan ke dalam tiga bagian yaitu :

1. Pencapaian tingkat kematangan 3
2. Pencapaian tingkat kematangan 4
3. Pencapaian tingkat kematangan 5

Proses pematangan atribut harus dilakukan secara bersama-sama sehingga semua atribut berada pada tingkat kematangan yang sama, kecuali untuk atribut Skill and Expertise yang hanya sampai pada tingkat 4.

5.4 Rekomendasi Perbaikan

Seperti yang telah dijelaskan sebelumnya rekomendasi tindakan perbaikan dibagi menjadi tiga kelompok agar proses peningkatan kematangan dapat berjalan lebih efektif sebagaimana pada gambar berikut :



Gambar 4.11 Pengelompokan Proses Rekomendasi perbaikan

5.5 Merumuskan Strategi Perbaikan

5.5.1 Pencapaian Tingkat Kematangan 3

Beberapa tindakan yang perlu dilakukan agar nilai kematangan pada semua atribut mencapai 3-*Defined* adalah sebagai berikut :

No	Atribut	Strategi Perbaikan
1	AC	<ul style="list-style-type: none"> • Mendefinisikan kebijakan perencanaan strategis TI • Menyelenggarakan pertemuan/rapat untuk membahas kebijakan perencanaan strategis TI • Mengkomunikasikan kebijakan ke seluruh unit kerja
2	PSP	<ul style="list-style-type: none"> • Mendokumentasikan kebijakan, standarisasi atau prosedur perencanaan strategis TI yang telah didefinisikan

3	TA	<ul style="list-style-type: none"> • Mendefinisikan alat bantu (missal SWOT) yang berkaitan dengan perencanaan strategis TI • Mendokumentasikan alat bantu yang berkaitan dengan perencanaan strategis TI
---	----	---

5.6 Indikator Kinerja dan Indikator Pencapaian

Sebagai tindak lanjut dari usulan perbaikan diatas, maka diperlukan pengukuran untuk mengetahui kemajuan yang dicapai. Pengukuran tersebut meliputi pelaksanaannya maupun pencapaiannya. Untuk itu perlu didefinisikan beberapa indikator pengukuran, yaitu Performance Indicator yang mengukur pelaksanaan dan Outcome Measure yang mengukur capaian hasil. Ada tiga aspek yang diukur yaitu :

1. Pencapaian dan kinerja TI (*IT Goals and Metrics*)
2. Pencapaian dan kinerja Proses (*Process Goals and Metrics*)
3. Pencapaian dan kinerja Aktifitas (*Activity Goals and Metrics*)

5.7 Perancangan Tata Kelola Perencanaan dan Organisasi TI

Perancangan model Tata Kelola Perencanaan dan Organisasi TI berangkat dari kebutuhan bisnis terkait dengan TI yang dijabarkan dalam visi misi Fakultas Teknik UWKS. Teknologi informasi yang dimiliki harus dapat mendukung dan sejalan dengan renstra sehingga responsive terhadap perubahan yang terjadi. Sehingga dalam hal ini untuk mencapai tujuan bisnis maka perlu diterjemahkan ke dalam IT Goals yaitu :

1. Menanggapi kebutuhan bisnis sejalan dengan strategi bisnis
2. Menanggapi kebutuhan tata kelola agar sejalan dengan arahan institusi
3. Memberikan kepuasan bisnis dari proyek dan portofolio aplikasi yang diterima Pencapaian IT Goals diatas dapat diukur dengan IT Metrics sebagai berikut :
 1. Tingkat persetujuan Institusi terhadap renstra TI
 2. Tingkat kesesuaian antara kebutuhan bisnis dan kebutuhan tata kelola
 3. Tingkat kepuasan bisnis dengan kondisi saat ini (jumlah, ruang lingkup, dll) dari proyek dan portofolio aplikasi

5.7.1 Rencana Aksi

Agar tujuan-tujuan yang telah ditetapkan diatas dapat segera tercapai secara efektif, maka diperlukan sebuah Rencana Aksi yang dikelompokkan menjadi tiga bagian yaitu :

1. Menanggapi kebutuhan bisnis sejalan dengan strategi bisnis
2. Menanggapi kebutuhan tata kelola agar sejalan dengan arahan institusi
3. Memberikan kepuasan bisnis dari proyek dan portofolio aplikasi yang diterima saat ini

6. Kesimpulan dan Saran

Dari hasil penelitian dapat ditarik beberapa kesimpulan sebagai berikut :

1. Pada pemilihan proses TI dengan menggunakan AHP, para responden memilih proses TI PO1 (Define Strategic IT Plan) dari lima pilihan proses TI (PO1, PO4, PO10, ME1, ME4)
2. Semua atribut proses TI yang terkait dengan Mendefinisikan Perencanaan Strategis TI saat ini berada pada tingkat kematangan 2 (Repeatable But Intuitive). Hal ini bahwa sudah terjadi proses yang berulang, namun masih tergantung pada pengetahuan individu sehingga kemungkinan terjadinya kesalahan masih cukup besar.
3. Sebagian besar atribut proses yang diharapkan berada di tingkat kematangan 5 (Optimized). Hal ini berarti pihak fakultas menginginkan proses yang telah disempurnakan ke tingkat yang lebih baik, serta menginginkan perbaikan yang terus menerus. TI digunakan secara terintegrasi untuk mengotomasi alur kerja, meningkatkan kualitas dan efektifitas.
4. Untuk atribut Skill and Expertise pada PO1 diharapkan untuk mencapai tingkat 4 (managed and measurable)
5. Dari hasil penelitian ini maka dihasilkan beberapa strategi perbaikan dan jadwal untuk rencana aksi

Masukan dan saran pada penelitian ini adalah :

1. Perlu adanya kedisiplinan untuk melaksanakan rencana aksi yang sesuai dengan jadwal agar hasil yang diharapkan dapat maksimal
2. Studi lebih lanjut dapat dilakukan untuk mengkaji proses PO4 Define the IT Process, PO10 Manage The Projects, ME1 Monitor and Evaluate IT Performance, ME4 Provide IT Governance yang terkait dengan Perencanaan dan Organisasi TI yang saat ini masih belum dilakukan karena keterbatasan sumber daya

3. Agar hasil tata kelola lebih maksimal, pihak fakultas pihak fakultas diharapkan dapat mengkombinasikan hasil penelitian serupa pada proses TI yang lain.

DAFTAR PUSTAKA

1. Cannon, David L., Bergmann, Timothy., Pamplin, Brady. 2006. *CISA Certified Information Systems Auditor, Study Guide*. United States of America: Sybex, Wiley Publishing, Inc
2. Cohn, John M., 2001. *Planning for integrated systems and technologies: a how-to-do-it manual for librarians*. New York: Neal-Schuman Publisher, Inc.
3. Edwards, Chris. 1995. *The Essence of Information Systems 2nd Edition*. United Kingdom: Prentice Hall International., Yogyakarta : Andi Publisier.
4. Gondodiyoto, Soenyoto. 2007. *Audit Sistem Informasi: Pendekatan Cobit, Edisi Revisi*. Jakarta: Mitra Wacana Media
5. Indrajit, Richardus Eko. 2000. *Pengantar Konsep Dasar Manajemen Sistem Informasi dan Teknologi Informasi*. Jakarta: PT Elex Media Komputindo.
6. Indrajit, Richardus Eko. 2004. *Kajian Strategis Cost Benefit Teknologi Informasi*. Yogyakarta: Penerbit Andi.
7. ITGI. 2005. *COBIT 4.0 : Control Objective, Management Guidelines, Maturity Models*. United States of America: IT Governance Institute.
8. Lancaster, F.W. 1988. *If you want to evaluate your library*. London: The Library Association.
9. Oetomo, Budi Sutejo Dharmo. 2002. *Perencanaan dan pengembangan sistem informasi*. Yogyakarta: Penerbit Andi
10. Umar, Husein, 2002. *Strategic management in action: konsep, teori, dan teknik menganalisa manajemen strategis strategic business unit berdasarkan konsep michael R. David, dan Wheelen-Hunger*. Jakarta: Gramedia.
11. Ward, John., Peppard, Joe. 2002. *Strategic Planning for Information System*. Cranfield, Bedfordshire, United Kingdom: John Wiley & Sons, LTD.

(5)

MENINGKATKAN KINERJA METODE DETEKSI TEPI FUZZY MENGGUNAKAN HIERARCHICAL CLUSTER ANALYSIS

Saiful Nur Budiman¹, Yanuar Risah Prayogi², Stendy B. Sakur³sync.saifulnb@gmail.com, yanuarrisah@gmail.com, stendy.sakur13@mhs.if.its.ac.id

ABSTRAK

Metode deteksi tepi *fuzzy* sebelumnya mampu mendeteksi tepi citra dengan histogram bimodal dengan lembah yang pendek. Bagaimanapun, untuk citra dengan histogram bimodal dengan lembah yang lebar, deteksi tepi yang dihasilkan belum memuaskan. Tujuan dari paper ini adalah untuk mendeteksi tepi pada citra dengan karakteristik histogram bimodal dengan lembah yang lebar. Metode yang diusulkan dikembangkan dari metode IFMFED (*improved fast multilevel fuzzy edge detection*). Hasil uji coba menunjukkan metode yang diusulkan dapat mendeteksi tepi pada citra dengan histogram bimodal dengan lembah yang lebar dan lebih baik dari metode IFMFED.

Kata Kunci: deteksi tepi, *fuzzy*, *hierarichal cluster analysis*, *threholding*.

I. PENDAHULUAN

Pemrosesan citra luas digunakan pada bidang industri misalnya untuk otomatisasi kontrol kualitas barang (*quality control*). Biasanya pengambilan citra menggunakan kamera berbasis CCD yang memungkinkan citra hasil kamera *blur* dan terdapat *noise*. Ketika dilakukan deteksi tepi pada citra *blur*, batas garis tepi menjadi tidak jelas. Deteksi tepi merupakan salah satu metode dasar pada pemrosesan citra. Kesulitan dalam deteksi tepi adalah ketika citra tersebut *blur* dan *noise* yang membuat batas garis tepi menjadi *fuzzy*.

Metode TFED (*Traditional Fuzzy Edge Detection*) adalah metode deteksi tepi yang berbasiskan *fuzzy* yang cocok untuk beberapa citra *blur* [4][6]. Pada metode TFED, pertama citra dirubah ke domain *fuzzy*. Kemudian nilai *fuzzy* tersebut ditingkatkan menggunakan sebuah operator *fuzzy*. Berikutnya dilakukan pengembalian dari domain *fuzzy* ke domain spasial. Dari domain spasial ini diekstrak garis tepinya menggunakan operator *min* dan *max*. Kelemahan algoritma ini adalah terlalu kompleks pada komputasinya yang eksponensial dan hilangnya piksel pada *grey level* rendah.

Penelitian berikutnya dilakukan oleh Jinbo Wu dkk [5] yang mengusulkan metode FMFED (*Fast Multilevel Fuzzy Edge Detection*) yang mampu melakukan deteksi tepi pada citra *blur* yang kompleks. Yang mana pada TFED tidak mampu melakukannya karena akan meningkatkan sebagian tepi pada citra dan melemahkan sebagian tepi yang lain. Metode FMFED menggabungkan antara

metode peningkatan *fuzzy* dengan dua tahap operator deteksi tepi. Kelebihan dari metode FMFED mampu meningkatkan kinerja deteksi tepi dalam hal *run time* dengan hasil yang lebih baik dari metode TFED. Namun metode ini tidak cocok untuk histogram bimodal dengan lembah yang pendek.

Untuk menutupi kekurangan dari FMFED, Xiangtao Chen dan Yujuan Chen [2] melakukan pengembangan terhadap metode FMFED menjadi IFMFED (*Improved Fast Multilevel Fuzzy Edge Detection*) dengan mengubah proses *threshold*, fungsi keanggotaan *fuzzy*, operator deteksi tepi, menambahkan median filtering, dan mengganti dua operator deteksi tepi dengan *fuzzy entropy*. Metode ini menggunakan algoritma *valley-emphasis* [3] untuk mendapatkan lembah dalam histogram yaitu nilai *threshold* tengah sedangkan *threshold* kiri dan kanan menggunakan rata-rata. Bagaimanapun, jika histogram citra bimodal dengan lembah yang lebar, metode IFMFED tidak bisa mendapatkan *threshold* kiri dan kanan yang optimal karena nilai rata-rata tidak bisa mengklaster histogram secara optimal.

Pada tulisan ini mengusulkan peningkatan metode deteksi tepi *fuzzy* menggunakan *hierarchical cluster analysis*. Metode yang diusulkan merupakan perbaikan dari metode IFMFED, dimana pada proses *threshold* menggunakan *hierarchical cluster analysis* [1] yang bisa mendapatkan tiga *threshold* sekaligus, *threshold* kiri, tengah, dan kanan sehingga membagi histogram menjadi 4 kluster yang optimal. Tulisan ini dibagi menjadi 4 bagian sebagai berikut: metode yang diusulkan pada bagian 2, hasil uji coba dan pembahasan pada bagian 3, dan kesimpulan pada bagian 4.

II. METODOLOGI

A. Analisis Algoritma IFMFED

Algoritma *IFMFED (Improved Fast Multilevel Fuzzy Edge Detection)* menggunakan metode *Valley-emphasis* [3] untuk mencari nilai *threshold* yang optimum T_m . Algoritma ini yang merupakan pengembangan dari metode Otsu dengan menambahkan nilai bobot sebesar $1 - P_t$, yaitu :

$$T_m = \text{ArgMax}_{0 \leq t \leq L} \{w_1(t)u_1^2(t) + w_2(t)u_2^2(t)(1 - p_t)\} \quad (1)$$

selanjutnya, merubahnya ke domain fuzzy menggunakan fungsi keanggotaan fuzzy yang berbentuk kurva konvex dengan persamaan berikut :

$$g_{x,y} = G'(f_{x,y}) = \begin{cases} 1 - \left(\frac{f_{x,y} - T_l}{T_l - f_{\min}}\right)^2, & f_{x,y} \leq T_l \\ 1 - \left(\frac{f_{x,y} - T_l}{T_l - T_m}\right)^2, & T_l < f_{x,y} \leq T_m \\ 1 - \left(\frac{f_{x,y} - T_h}{T_h - T_m}\right)^2, & T_m < f_{x,y} \leq T_h \\ 1 - \left(\frac{f_{x,y} - T_h}{T_h - T_m}\right)^2, & f_{x,y} > T_h \end{cases} \quad (2)$$

dimana, $g_{x,y}$ merupakan fungsi keanggotaan baru, T_m nilai *threshold* berdasarkan *valley-emphasis*. T_l dan T_h , merupakan nilai rata-rata dari *gray* terendah B_1 dan *gray* tertinggi B_2 yang dihitung secara berurutan sebagai berikut:

$$T_l = \frac{\sum_{f_{x,y} \in B_1} f_{x,y}}{B_1} \quad (3)$$

$$T_h = \frac{\sum_{f_{x,y} \in B_2} f_{x,y}}{B_2} \quad (4)$$

dimana, B_1 dan B_2 menunjukkan pixel obyek dan pixel latar belakang. Selanjutnya dilakukan proses *median filter* untuk mengurangi *noise* pada citra dengan persamaan:

$$g'_{x,y} = \frac{1}{8}(g_{x-1,y-1} + g_{x,y-1} + g_{x+1,y-1} + g_{x-1,y} + g_{x+1,y} + g_{x-1,y+1} + g_{x,y+1} + g_{x+1,y+1}) \quad (5)$$

Selanjutnya dilakukan peningkatan nilai fuzzy menggunakan operator *fuzzy enhancement* yang ditunjukkan pada persamaan:

$$I_r = I_1(I_{r-1}(g'_{x,y})) \quad (6)$$

dimana

$$I_1(g'_{x,y}) = \begin{cases} 2(g'_{x,y})^2 & 0 \leq g'_{x,y} \leq t \\ 1 - 2(1 - g'_{x,y})^2 & t < g'_{x,y} \leq 1 \end{cases}$$

dengan nilai $t = 0.5$ dan $r = 3$.

Proses selanjutnya melakukan transformasi dari domain *fuzzy* ke domain spasial dengan melakukan *invers* dari G' . Langkah terakhir yang digunakan adalah mengukur nilai entropi dari nilai *gray* untuk seluruh piksel dengan menggunakan persamaan berikut :

$$E_m(A) = \frac{1}{9} \sum_{i=x-1}^{i=x+1} \sum_{j=y-1}^{j=y+1} (H_m(\omega_m(i, j))) \quad (7)$$

dimana,

$$H_m(\omega_m(i, j)) = -\omega_m(i, j) \log(\omega_m(i, j)) - (1 - \omega_m(i, j)) \log(1 - \omega_m(i, j))$$

dengan

$$\omega_m(i, j) = \frac{1}{1 + \frac{|g'_{i,j} - m|}{C}}$$

pada persamaan diatas, m adalah nilai tengah dari *gray level* (x, y) dari *windows* berukuran 3×3 dan nilainya tetap. Dengan adanya hubungan antara $g'_{x,y}$ dan $E_m(A)$ dapat di tentukan tepi $b_{x,y}$, dengan persamaan berikut,

$$b_{x,y} = \begin{cases} 1, & E_{m(x,y)}(A) \geq Q \\ 0, & E_{m(x,y)}(A) \leq Q \end{cases} \quad (8)$$

dimana Q merupakan nilai *threshold* yang telah ditentukan sebelumnya.

B. Peningkatan Algoritma Deteksi Tepi Fuzzy

Menggunakan hierarchical cluster analysis untuk penentuan threshold

Metode *IFMFED* menggunakan *Valley-emphasis* untuk menentukan nilai *threshold* yang optimum untuk histogram dengan distribusi normal seperti terlihat pada Gambar 1, pada kasus seperti gambar 2, metode *IFMFED* dapat menentukan nilai *threshold* T_m secara optimal akan tetapi dua nilai rata-rata yaitu T_l dan T_h tidak dapat secara optimal di tentukan.

Metode yang diusulkan menggunakan *hierarchical cluster analysis* [1] untuk menentukan nilai *threshold* yang optimal dengan menggunakan strategi penggabungan nilai kluster berdasarkan pada perbedaan rata-rata dari dua kluster dan hasil dari variannya. Misalkan, $y(t)$ dengan $t = 0, 1, 2, \dots, L-1$ merupakan histogram dari citra tujuan dimana t merupakan *gray level* dan L jumlah dari *gray level* termasuk nilai 0. Histogram $y(t)$ berupa akurasi frekuensi dari piksel dengan *gray level* t . Jika di

definisikan $r(t)$ sebagai $r(t) = y(t)/N$, dimana N adalah jumlah dari piksel pada citra dengan $y(t)$ dimisalkan sebagai probabilitas dari akurasi piksel dengan *gray level* t . Fungsi cluster $Y(C_k)$ didefinisikan sebagai :

$$Y(C_k) = \sum_{t=T_{k-1}+1}^{T_k} r(t), \quad \sum_{k=1}^K Y(C_k) = 1 \quad (9)$$

fungsi ini mengindikasikan sebagai akurasi probabilitas dari piksel pada cluster C_k . Jarak antara cluster C_{k1} dan C_{k2} didefinisikan sebagai,

$$Dist(C_{k1}, C_{k2}) = \sigma_1^2(C_{k1} \cup C_{k2}) \sigma_A^2(C_{k1} \cup C_{k2}) \quad (10)$$

dimana kedua paramater tersebut adalah *inter-class variance* dan *intra-class variance*. *Inter-class variance*, $\sigma_1^2(C_{k1}, C_{k2})$, merupakan penjumlahan dari akar kuadrat jarak dua cluster dan total rata-rata dua cluster didefinisikan sebagai berikut :

$$\sigma_1^2(C_{k1}, C_{k2}) = \frac{Y(C_{k1})Y(C_{k2})}{(Y(C_{k1})+Y(C_{k2}))^2} [m(C_{k1}) - m(C_{k2})]^2 \quad (11)$$

Dimana $m(C_k)$ merupakan rata-rata dari cluster C_k yang didefinisikan sebagai berikut,

$$m(C_k) = \frac{1}{Y(C_k)} \sum_{t=T_{k-1}+1}^{T_k} tr(t) \quad (12)$$

sedangkan *intra-class variance*, $\sigma_A^2(C_{k1} \cup C_{k2})$ didefinisikan sebagai berikut,

$$\sigma_A^2(C_{k1} \cup C_{k2}) = \frac{1}{Y(C_{k1}) + Y(C_{k2})} \times \sum_{t=T_{k1-1}+1}^{T_{k2}} [(t - M(C_{k1} \cup C_{k2}))^2 r(t)] \quad (13)$$

dari perhitungan diatas di peroleh nilai *threshold* optimal T_m dan dua nilai *threshold* T_l dan T_h .

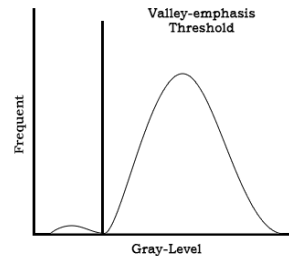
Prosedur pada metode yang di usulkan

Algoritma yang diusulkan menggunakan *hierarchical cluster analysis* untuk menentukan nilai *threshold* yang optimal. Prosedur perhitungan untuk metode yang di usulkan sebagai berikut:

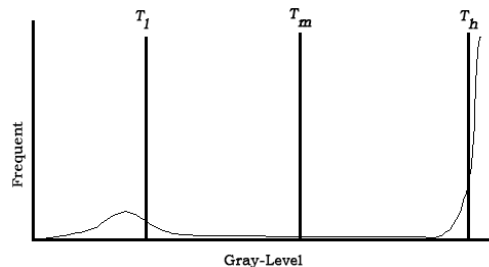
1. Mendapatkan nilai *threshold* T_l , T_m , dan T_h menggunakan *hierarchical cluster analysis*.
2. Mendefinisikan fungsi keanggotaan fuzzy menggunakan tiga *threshold* pada langkah 1. Dari fungsi keanggotaan fuzzy pada persamaan 2, domain sepasial citra $f_{x,y}$ dirubah ke domain fuzzy $g_{x,y}$.
3. Mendapatkan domain fuzzy yang sudah dihaluskan $g'_{x,y}$ menggunakan median filter.
4. Meningkatkan domain fuzzy menggunakan operator $I_r(g'_{x,y})$ dengan nilai $t = 0.5$ dan $r = 3$.
5. Mengembalikan dari domain fuzzy ke domain sepasial menggunakan fungsi invers dari $G'(f_{x,y})$.

6. Mendapatkan binari citra dengan menggunakan operator fuzzy entropi yang ditunjukkan pada persamaan 7 dan 8 dengan nilai $Q = 0.1$.

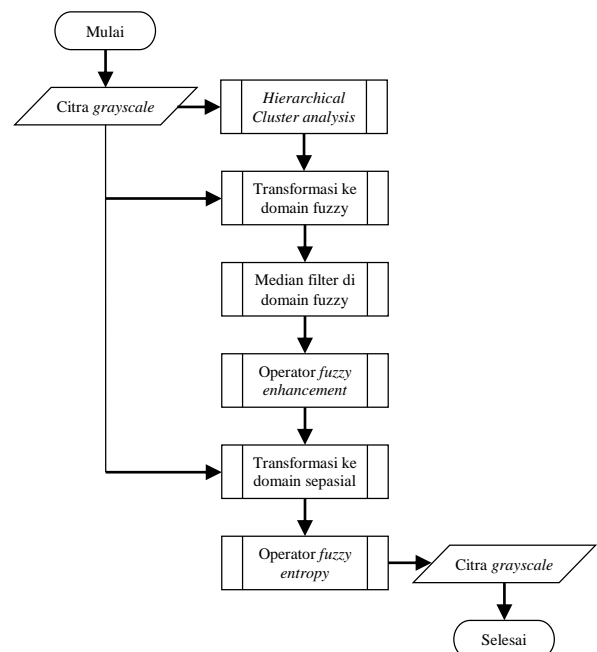
Gambar langkah-langkah pada metode yang diusulkan ditunjukkan pada Gambar 3.



Gambar 1. Unimodal yang dideteksi



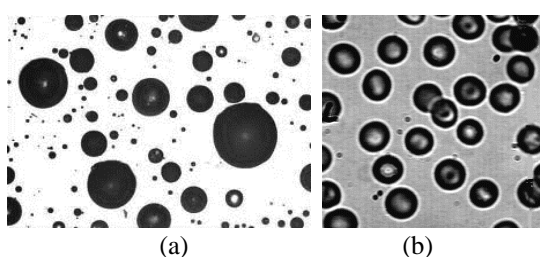
Gambar 2. Tidak Optimal untuk IFMFED



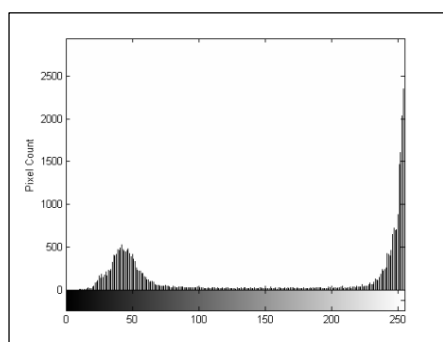
Gambar 3. Langkah-langkah pada metode yang diusulkan

III. HASIL DAN PEMBAHASAN

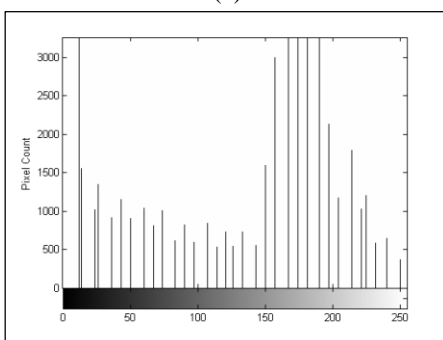
Pada bagian ini akan dipaparkan hasil percobaan menggunakan algoritma yang diusulkan. Citra yang digunakan sebagai uji coba ada dua citra yang ditunjukkan pada gambar 4. Histogram dari gambar uji coba adalah histogram bimodal dengan lembah yang lebar seperti yang ditunjukkan pada gambar 5. Dalam rangka untuk mengevaluasi hasil kinerja, metode yang diusulkan akan dibandingkan dengan algoritma IFMFED (*improved fast multilevel fuzzy edge detection*). Sedangkan tingkat kinerja akan diukur menggunakan *precision*, *sensitivity*, dan *accuracy*.



Gambar 4. Citra uji coba (a) Citra enamel gigi (b) Citra sel darah merah



(a)

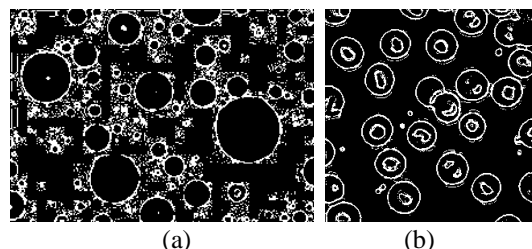


(b)

Gambar 5. Histogram citra uji coba (a) Citra enamel gigi (b) Citra sel darah merah

Hasil dari deteksi tepi menggunakan IFMFED ditunjukkan pada gambar 6 sedangkan hasil deteksi tepi dari metode yang diusulkan ditunjukkan pada

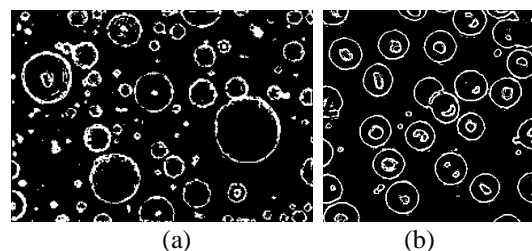
gambar 7. Pada gambar 6 hasil ekstraksi tepi tidak bersih dan masih ada yang bukan tepi tetapi dianggap sebagai tepi. Hasil yang tidak bersih ini disebabkan oleh pengaruh derau. Sedangkan pada metode yang diusulkan, garis tepi yang dihasilkan bagus sekaligus mengurangi derau seperti yang ditunjukkan pada gambar 7.



(a)

(b)

Gambar 6. Hasil deteksi tepi menggunakan IFMFED (a) Citra enamel gigi (b) Citra sel darah merah



(a)

(b)

Gambar 7. Hasil deteksi tepi menggunakan metode yang diusulkan (a) Citra enamel gigi (b) Citra sel darah merah

Nilai kinerja *precision*, *sensitivity*, dan *accuracy* tergantung dari nilai *true positive* (TP), *false positive* (FP), *false negative* (FN), dan *true negative* (TN). Persamaan *precision*, *sensitivity*, dan *accuracy* secara berurutan didefinisikan sebagai

$$precision = TP / (TP + FP), \quad (14)$$

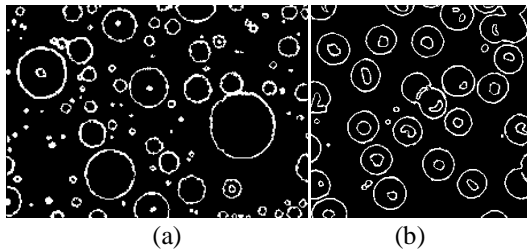
$$sensitivity = TP / (TP + FN), \quad (15)$$

$$accuracy = (TP + TN) / (Total \text{ pixel}), \quad (16)$$

dimana *total pixel* adalah jumlah pixel yang ada pada citra. Untuk mengukur kinerja *precision*, *sensitivity*, dan *accuracy* harus dibandingkan dengan citra *ground-truth* yang ditunjukkan pada gambar 8. Tabel kinerja metode IFMFED ditunjukkan pada tabel 1 sedangkan tabel kinerja metode yang diusulkan ditunjukkan pada tabel 2.

Pada metode IFMFED penentuan *threshold* tengah menggunakan algoritma *valley-emphasis* sedangkan dua *threshold*, *threshold* kiri dan kanan, menggunakan rata-rata. Metode IFMFED berhasil menentukan *threshold* tengah tetapi tidak untuk *threshold* kiri dan kanan sehingga tidak dapat

membagi histogram menjadi empat kluster yang optimal. Gambar *threshold* pada histogram citra dari metode IFMFED dan metode yang diusulkan ditunjukkan pada gambar 9.



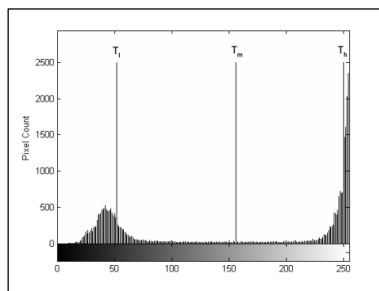
Gambar 8. Citra ground-truth (a) Citra enamel gigi (b) Citra sel darah merah

Tabel 1. Kinerja metode IFMFED

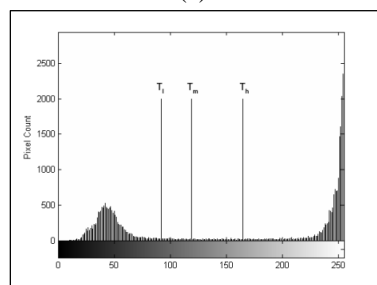
Citra uji coba	<i>precision</i>	<i>sensitivity</i>	<i>accuracy</i>
Enamel gigi	0.35	0.63	0.84
Sel darah merah	0.66	0.85	0.94

Tabel 2. Kinerja metode yang diusulkan

Citra uji coba	<i>precision</i>	<i>sensitivity</i>	<i>accuracy</i>
Enamel gigi	0.83	0.95	0.98
Sel darah merah	0.87	0.98	0.98



(a)



(b)

Gambar 9. *Threshold* pada histogram citra enamel gigi (a) metode IFMFED (b) Metode yang diusulkan

Pada gambar 9 (a) dapat kita lihat bahwa *threshold* kanan letaknya terlalu disebelah kanan. Metode yang diusulkan menggunakan *hierarchical cluster analysis* yang mampu menentukan tiga *threshold* sekaligus, *threshold* kiri, tengah, dan kanan, menjadi empat kluster yang optimal seperti yang ditunjukkan pada gambar 9 (b).

IV. KESIMPULAN

Algoritma IFMFED tidak dapat mengekstrak garis tepi pada citra dengan karakter histogram bimodal dengan lembah yang lebar. Hasil dari deteksi tepi IFMFED masih mendeteksi *false edge* dan *discontinuous*. Hasil dari IFMFED masih terdapat derau sehingga hasilnya tidak bersih. Dari tabel kinerja didapatkan bahwa metode yang diusulkan memiliki kinerja lebih tinggi dari metode IFMFED. Metode yang diusulkan ini dapat menutupi kekurangan dari IFMFED. Metode yang diusulkan dapat mengekstrak garis tepi yang *continuous* dan mengurangi *noise*. Dari hasil eksperimen menunjukkan bahwa metode yang diusulkan dapat dijadikan sebagai deteksi tepi pada citra dengan karakter histogram bimodal dengan lembah yang lebar.

V. DAFTAR PUSTAKA

1. Arifin, A.Z. and A. Asano, *Citra Segmentation by Histogram Thresholding Using Hierarchical Cluster Analysis*. 2006.
2. Chen, X. and Y. Chen, *An Improved Edge Detection in Noisy Citra Using Fuzzy Enhancement*. 2010.
3. Ng, H.-F., *Automatic Thresholding for Defect Detection*. Proceeding of the Third International Conference on Citra and Graphics (ICIG'04), 2004. 0-7695-2244-0/04.
4. S. K. Pal and R. A. King, "On edge detection of X-ray images using fuzzy sets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-5, no.1, pp. 69–77, Jan. 1983.
5. Wu, J., Z. Yan, and Y. Xiong, *The Fast Multilevel Fuzzy Edge Detection of Blurry Citras*. *IEEE Signal Processing Letters*, 2007. 14.No. 5, May 2007.
6. Y. Nakagawa and Y. Rosenfeld, "A note on the use of local min and max operations in digital picture processing," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, no. 8, pp. 632–635, Aug. 1978.

{Halaman sengaja dikosongkan}
{Jurnal **Melek IT**}

(3)

APLIKASI PENGHITUNGAN PAJAK PENGHASILAN BERBASIS ANDROID DENGAN STUDI KASUS RSUD BHAKTI DHARMA HUSADA SURABAYA**Fajar Wahyudi¹, Anang Kukuh Adisusilo ST.MT²**

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Wijaya Kusuma Surabaya

fajar.tif037@gmail.com, anang@anang65.web.id**ABSTRAK**

Beberapa karyawan di suatu rumah sakit telah menggunakan aplikasi perhitungan PPh.21 yang terintegrasi pada aplikasi *payroll*, sedangkan lainnya masih mengerjakan secara manual atau menggunakan *Spread Sheet*. Baik perusahaan yang telah menggunakan *software* aplikasi *payroll* untuk menghitung PPh.21 terutangannya, maupun yang masih mengerjakannya secara manual mengalami kendala berupa kesalahan dalam menentukan besarnya PPh.21 terutang atau jumlah total angsurannya kurang dari jumlah PPh.21 terutang.

Untuk itu dibutuhkan suatu sistem dengan yang tepat sehingga secara akurat dapat memberikan jawaban terhadap kendala yang sering dihadapi perusahaan dalam menghitung besarnya PPh.21 terutang. Penghitungan Pajak Penghasilan Pasal 21 perorangan dirancang dalam sebuah aplikasi berbasis android akan memudahkan seorang karyawan mengetahui berapa pajak yang harus dibayarkan dan tentunya dengan penghitungan pph 21 yang telah berlaku.

Kata kunci : Penghitungan, Pajak Penghasilan Pasal 21, Android**PENDAHULUAN****Latar Belakang**

Dalam suatu Rumah Sakit tidak luput dari pengelolaan masalah administrasi, karena administrasi merupakan titik pokok di dalamnya. Untuk berkembang, tumbuh dan bergerak rumah sakit juga membutuhkan administrasi yang baik. Administrasi adalah sesuatu yang terdapat di dalam organisasi yang moderen yang memberikan manfaat dari organisasi tersebut, sehingga organisasi itu dapat berkembang, tumbuh dan bergerak. Pajak merupakan iuran wajib masyarakat kepada negara yang dapat dipaksakan tanpa mendapat kontraprestasi secara langsung, dan apabila ada bagian masyarakat yang tidak melunasinya maka akan dikenakan sanksi oleh negara.

Pada hakekatnya setiap karyawan yang bekerja di Rumah Sakit memiliki potensi untuk menjadi wajib pajak. Dan yang menjadi obyek pajak adalah atas penghasilan yang diperoleh dan diterima oleh masyarakat. Pengertian dari penghasilan adalah setiap tambahan kemampuan ekonomis yang diterima atau diperoleh Wajib Pajak, baik yang berasal dari Indonesia maupun dari luar Indonesia, yang dapat dipakai untuk konsumsi atau menambah kekayaan Wajib Pajak yang bersangkutan, dengan nama dan bentuk apapun. Pajak yang dikenakan atas penghasilan yang diterima atau diperoleh Wajib Pajak disebut Pajak penghasilan. Pengertian dari Pajak

Penghasilan sendiri itu adalah suatu pungutan resmi yang ditujukan kepada masyarakat yang berpenghasilan atau atas penghasilan yang diterima atau diperolehnya dalam tahun pajak untuk kepentingan negara dan masyarakat.

Salah satu jenis pajak penghasilan adalah PPh Pasal 21. PPh pasal 21 merupakan pajak terutang atas penghasilan yang menjadi kewajiban bagi wajib pajak orang pribadi yang bekerja pada satu pemberi kerja (pegawai / karyawan) untuk membayarnya. Penghasilan yang dimaksud disini adalah penghasilan yang berupa penggantian atau imbalan berkenaan dengan pekerjaan atau jasa yang diterima atau diperoleh termasuk gaji, upah, tunjangan, honorarium, komisi, bonus, gratifikasi, uang pensiun, atau imbalan dalam bentuk lainnya. Disebut PPh Pasal 21 karena ketentuan perpajakan yang berkenaan dengan penghasilan karyawan itu diatur dalam Undang-Undang Pajak Penghasilan No. 17 Tahun 2000 pasal 21.

Pada Rumah Sakit Umum Daerah Bhakti Dharma Husada (BDH) Surabaya masih menggunakan aplikasi perhitungan PPh.21 yang terintegrasi pada aplikasi *payroll*, sedangkan lainnya masih mengerjakan secara manual atau menggunakan *Spread Sheet*. Baik perusahaan yang telah menggunakan *software* aplikasi *payroll* untuk menghitung PPh.21 terutangannya, maupun yang masih mengerjakannya secara manual mengalami kendala berupa kesalahan dalam menentukan besarnya PPh.21 terutang atau jumlah total angsurannya kurang dari jumlah PPh.21 terutang.

Untuk itu dibutuhkan suatu sistem dengan yang tepat sehingga secara akurat dapat memberikan jawaban terhadap kendala yang sering dihadapi rumah sakit dalam menghitung besarnya PPh.21 terutang.

Penghitungan Pajak Penghasilan Pasal 21 perorangan dirancang dalam sebuah aplikasi berbasis android akan memudahkan seorang karyawan mengetahui berapa pajak yang harus dibayarkan dan tentunya dengan penghitungan pph 21 yang telah berlaku.

TINJAUAN PUSTAKA

Definisi Pajak Penghasilan Pasal 21

Pajak Penghasilan Pasal 21 adalah pajak atas penghasilan berupa gaji, upah, honorarium, tunjangan dan pembayaran lainnya sehubungan dengan pekerjaan atau jabatan, jasa, dan kegiatan lainnya yang dilakukan oleh WP Orang Pribadi Dalam Negeri.

Pemotong Pajak, yaitu:

1. **Pemberi kerja** yang terdiri dari *orang pribadi* dan *badan*, baik merupakan pusat maupun cabang, perwakilan atau unit, bentuk usaha tetap, yang membayar gaji, upah, honorarium, tunjangan, dan pembayaran lain dengan nama apapun, sebagai imbalan sehubungan dengan pekerjaan atau jasa yang dilakukan oleh pegawai atau bukan pegawai.
2. **Bendaharawan Pemerintah.**
3. **Dana pensiun**, badan penyelenggara **Jamsostek**, dan badan-badan lain yang membayar uang pensiun dan Tabungan Hari Tua atau Jaminan Hari Tua.
4. Orang pribadi yang melakukan kegiatan usaha atau pekerjaan bebas yang melakukan pembayaran honorarium.
5. **Penyelenggara kegiatan** (termasuk badan pemerintah, organisasi termasuk organisasi internasional, perkumpulan, orang pribadi serta lembaga lainnya yang menyelenggarakan kegiatan) yang membayar honorarium, hadiah atau penghargaan dalam bentuk apapun kepada Wajib Pajak orang pribadi dalam negeri berkenaan dengan suatu kegiatan.

Dikecualikan dari Pemotong Pajak :

1. Badan Perwakilan Negara Asing.
2. Organisasi Internasional.
3. Orang pribadi yang tidak melakukan usaha atau pekerjaan bebas dan semata-mata mempekerjakan orang pribadi untuk

pekerjaan rumah tangga atau pekerjaan bukan dalam lingkup usaha.

2.1. Tarif dan Penerapan Penghitungan Pph Pasal 21

1. Pegawai tetap, penerima pensiun bulanan, bukan pegawai yang memiliki NPWP dan menerima penghasilan secara berkesinambungan dalam 1 (satu) tahun dikenakan tarif Pasal 17 ayat (1) huruf a Undang-Undang PPh dikalikan dengan Penghasilan Kena Pajak (PKP). PKP dihitung berdasarkan sebagai berikut:
 - a. Pegawai Tetap: Penghasilan bruto dikurangi biaya jabatan (5% dari penghasilan bruto, maksimum Rp 6.000.000,00 setahun atau Rp 500.000,00 sebulan); dikurangi iuran pensiun, Iuran jaminan hari tua, dikurangi Penghasilan Tidak Kena Pajak (PTKP).
 - b. Penerima Pensiun Bulanan: Penghasilan bruto dikurangi biaya pensiun (5% dari penghasilan bruto, maksimum Rp 2.400.000,00 setahun atau Rp 200.000,00 sebulan) dikurangi PTKP.
 - c. Bukan Pegawai yang memiliki NPWP dan menerima penghasilan secara berkesinambungan: 50 % dari Penghasilan bruto dikurangi PTKP perbulan.
2. Bukan Pegawai yang menerima atau memperoleh penghasilan dikenakan tarif Pasal 17 ayat (1) huruf a dikalikan dengan 50% dari jumlah penghasilan bruto untuk setiap pembayaran imbalan yang tidak berkesinambungan;
3. Peserta kegiatan yang menerima atau memperoleh penghasilan dikenakan tarif Pasal 17 ayat (1) huruf a dikalikan dengan jumlah penghasilan bruto untuk setiap kali pembayaran yang bersifat utuh dan tidak dipecah;
4. Pegawai harian, pegawai mingguan, pemegang, dan calon pegawai, serta pegawai tidak tetap lainnya yang menerima upah harian, upah mingguan, upah satuan, upah borongan dan uang saku harian yang besarnya melebihi Rp.150.000 sehari tetapi dalam satu bulan takwim jumlahnya tidak melebihi Rp. 1.320.000,00 dan atau tidak dibayarkan secara bulanan, maka PPh Pasal 21 yang terutang dalam sehari adalah dengan menerapkan tarif 5% dari penghasilan bruto setelah dikurangi Rp. 150.000,00. Bila dalam satu bulan takwim jumlahnya melebihi Rp.1.320.000,00 sebulan, maka besarnya PTKP yang dapat dikurangkan untuk satu hari adalah sesuai dengan jumlah

PTKP sebenarnya dari penerima penghasilan yang bersangkutan dibagi 360.

5. Pejabat Negara, PNS, anggota TNI/POLRI yang menerima honorarium dan imbalan lain yang sumber dananya berasal dari Keuangan Negara atau Keuangan Daerah dipotong PPh Ps. 21 dengan tarif 15% dari penghasilan bruto dan bersifat final, kecuali yang dibayarkan kepada PNS Gol. IId kebawah, anggota TNI/POLRI Peltu kebawah/ Ajun Insp./Tingkat I kebawah.
Besarnya PTKP adalah :

Penerima PTKP	Setahun	Sebulan
untuk diri pegawai	Rp 15.840.000	Rp 1.320.000
tambahan untuk pegawai yang sudah menikah(kawin)	Rp 1.320.000	Rp 110.000
tambahan untuk setiap anggota keluarga *) paling banyak 3 (tiga) orang	Rp 1.320.000	Rp 110.000

6. *) anggota keluarga adalah anggota keluarga sedarah dan semenda dalam satu garis keturunan lurus, serta anak angkat yang menjadi tanggungan sepenuhnya.
7. Tarif Pasal 17 ayat (1) huruf a Undang-undang Pajak Penghasilan adalah:

Lapisan Penghasilan Kena Pajak	Tarif
sampai dengan Rp 50 juta	5%
diatas Rp 50 juta sampai dengan Rp 250 juta	15%
diatas Rp 250 juta sampai dengan Rp 500 juta	25%
diatas Rp 500 juta	30%

8. Bagi Wajib Pajak yang tidak memiliki NPWP dikenakan tarif 20 % lebih tinggi dari tarif PPh Pasal 17.

UML

UML (Unified Modeling Language) adalah sebuah bahasa untuk menentukan, visualisasi, konstruksi, dan mendokumentasikan artifact (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan perangkat lunak. Artifact dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya.

UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

Bagian-bagian UML

Bagian-bagian utama dari UML adalah view, diagram, model element, dan general mechanism.

2.3.1 View

View digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. View bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram.

Beberapa jenis view dalam UML antara lain: use case view, logical view, component view, concurrency view, dan deployment view.

1. Use case view

Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan external actors. Actor yang berinteraksi dengan sistem dapat berupa user atau sistem lainnya.

View ini digambarkan dalam use case diagrams dan kadang-kadang dengan activity diagrams. View ini digunakan terutama untuk pelanggan, perancang (designer), pengembang (developer), dan pengujian sistem (tester).

2. Logical view

Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis (class, obyek, dan relationship) dan kolaborasi dinamis yang terjadi ketika obyek mengirim pesan ke obyek lain dalam suatu fungsi tertentu.

View ini digambarkan dalam class diagrams untuk struktur statis dan dalam state, sequence, collaboration, dan activity diagram untuk model dinamisnya. View ini digunakan untuk perancang (designer) dan pengembang (developer).

3. Component view

Mendeskripsikan implementasi dan ketergantungan modul. Komponen yang merupakan tipe lainnya dari code module diperlihatkan dengan struktur dan ketergantungannya juga alokasi sumber daya komponen dan informasi administrative lainnya. View ini digambarkan dalam component view dan digunakan untuk pengembang (developer).

4. Concurrency view

Membagi sistem ke dalam proses dan prosesor. View ini digambarkan dalam diagram dinamis (state, sequence, collaboration, dan activity diagrams) dan diagram implementasi (component dan deployment diagrams) serta digunakan untuk pengembang (developer), pengintegrasian (integrator), dan pengujian (tester).

5. Deployment view

Mendeskripsikan fisik dari sistem seperti komputer dan perangkat (nodes) dan bagaimana hubungannya dengan lainnya. View ini digambarkan dalam deployment diagrams dan digunakan untuk pengembang (developer), pengintegrasian (integrator), dan pengujian (tester).

Diagram

Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu view tertentu dan ketika digambarkan biasanya dialokasikan untuk view tertentu. Adapun jenis diagram antara lain :

- a. Use case Diagram
- b. Class Diagram
- c. Component Diagram
- d. Deployment Diagram
- e. State Diagram
- f. Sequence Diagram
- g. Collaboration Diagram
- h. Activity Diagram

Tujuan Penggunaan UML.

- i. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
- ii. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.
- iii. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.

UML bisa juga berfungsi sebagai sebuah (blue print) cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bias diketahui informasi secara detail tentang koding program atau bahkan membaca program dan menginterpretasikan kembali ke dalam bentuk diagram (reverse engineering).

Perangkat lunak yang mendukung pembuatan diagram UML :

StarUML

(<http://staruml.sourceforge.net/en/>)

StarUML adalah sebuah proyek open source untuk mengembangkan cepat, fleksibel, extensible, featureful, dan bebas-tersedia UML / platform MDA berjalan pada platform Win32. Tujuan dari proyek StarUML adalah untuk membangun sebuah alat pemodelan perangkat lunak dan juga platform yang menarik adalah pengganti alat UML komersial seperti Rational Rose, Bersama dan sebagainya.

Android

Mengacu pada Bhawiyuga, 2011, Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan *platform* terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti bergerak. Android merupakan

sebuah sistem operasi seperti halnya symbian pada Nokia, Palm, dan Windows Mobile yang sebelumnya sudah terlebih dahulu kita kenal selama ini. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance* konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau *Google Mail Services* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD). Sistem perangkat lunak Android sebenarnya merupakan sebuah *stack* atau tumpukan yang terdiri atas beberapa lapisan dari yang paling dekat dengan perangkat keras sampai yang berinteraksi langsung dengan pengguna. Jadi, dapat dikatakan bahwa Android bukan sekedar suatu *framework* yang berdiri di atas suatu sistem saja, tapi Android merupakan sistem itu sendiri yang tersusun atas berbagai lapisan perangkat lunak.

Dalam pembuatan aplikasi, diperlukan beberapa instalasi *software* yang mendukung antara lain Java, Android SDK, Eclipse, dan ADT.

2.4.1. Java Platform, Standard Edition (Java SE)

Menurut Kadir, 2004, Java adalah bahasa pemrograman serbaguna. Java dapat digunakan untuk membuat suatu program sebagaimana anda membuatnya dengan bahasa seperti Pascal atau C++. Yang lebih menarik, java juga mendukung sumber daya internet yang saat ini populer, yaitu *World Wide Web* atau yang sering disebut *Web* saja. Java juga mendukung aplikasi klien/server, baik dalam jaringan local (LAN) maupun jaringan berskala luas (WAN). Java dikembangkan oleh Sun Microsystems pada Agustus 1991, dengan nama semula Oak. Konon Oak adalah pohon semacam jati yang terlihat dari jendela tempat pembuatnya, James Gosling, bekerja. Ada yang mengatakan bahwa Oak adalah singkatan dari “Object Application Kernel”, tetapi ada yang menyatakan hal itu muncul setelah nama Oak diberikan. Pada Januari 1995, Karena nama Oak dianggap kurang komersial, maka diganti Java (Kadir, 2004).

Dalam sejumlah literatur disebutkan bahwa Java merupakan hasil perpaduan sifat dari sejumlah bahasa pemrograman, yaitu C, C++, Object-C, SmallTalk, dan Common LISP. Selain itu java juga dilengkapi dengan unsur keamanan.

Yang tak kalah penting adalah Java menambahkan paradigma pemrograman yang sederhana. Jika anda telah mengenal C atau C++, yang mengandalkan *pointer* dan anda dapat merasakan keruwetannya, Java justru meninggalkannya sehingga anda akan memperoleh kemudahan saat menggunakannya (Kadir, 2004). Mengacu pada Kadir, 2004, program Java bersifat tidak bergantung pada *platform*, artinya, Java dapat dijalankan pada sebarang komputer dan bahkan pada sebarang sistem operasi.

Ketidaktergantungan terhadap *platform* sering dinyatakan dengan istilah portabilitas. Yang menarik, tingkat portabilitas Java tidak hanya sebatas pada program sumber (*source code*), melainkan juga pada tingkat kode biner yang disebut *bytecode*. Dengan demikian bila anda telah mengkompilasi program Java pada komputer bersistem operasi Windows, anda dapat menjalankan hasil kompilasi pada Macintosh secara langsung, tanpa perlu mengkompilasi ulang. Kode yang disebut *bytecode* dapat dijalankan pada berbagai sistem operasi karena kode ini berbeda dengan kode mesin. Kode mesin sangat bergantung pada *platform*, sedangkan *bytecode* dapat dimengerti oleh semua *platform* yang telah dilengkapi dengan interpreter Java. Mengingat bahwa hasil kompilasi Java dapat dijalankan pada sebarang sistem operasi ataupun prosesor, Java sering dikatakan bersifat netral terhadap arsitektur komputer (Kadir, 2004).

Android SDK (Software Development Kit)

Menurut Safaat, 2011, Android SDK adalah *tools API (Application Programming Interface)* yang diperlukan untuk memulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Saat ini disediakan Android SDK sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java.

Eclipse

Menurut Pisa, 2009, Eclipse adalah sebuah pengembangan lingkungan yang terintegrasi yang membantu Anda untuk membangun aplikasi-aplikasi dalam banyak bahasa. Eclipse adalah *software open source* yang didukung oleh Eclipse *foundation*, sebuah asosiasi *non profit* yang serupa dalam ruang lingkup dasar Apache.

Eclipse pada awalnya dikembangkan oleh OTI (*Object technology International*) dan kemudian oleh IBM sebagai bagian dari proyek VisualAge. Hal ini kemudian menjadi bagian dari strategi IBM untuk membuat kode *open source*

untuk mendorong adaptasi dari teknologi-teknologi yang dirancang untuk berjalan pada *platform* yang lebih disukai oleh IBM. Berkat arsitekturnya yang sangat kuat dan stabil, dan karena sebagian yang ditulis dalam bahasa Java, Eclipse dapat berjalan di beberapa sistem operasi dengan mudah. Inti dari Eclipse adalah *plug-in manager* yang dapat memuat, menginstal, dan menjalankan berbagai *plug-in*, masing-masing seperti komponen yang menyediakan berbagai jenis layanan. Terdapat *plug-in* yang mengetes, *debug*, dan menjalankan kode Anda, dan membantu Anda menulis kode dalam banyak bahasa yang berbeda, seperti AS, JavaScript, CF, PHP dan lain-lain. Bahasa *default*-nya adalah selalu Java.

ADT (Android Development Tools)

Menurut Safaat, 2011, ADT adalah *plug-in* yang membuat Eclipse dapat membuat *project* berbasis Android. ADT harus di-*install*, karena sebagai penghubung antara Android SDK dengan IDE Eclipse yang akan digunakan sebagai tempat *coding* aplikasi Android nantinya.

PhoneGap

PhoneGap adalah sebuah development framework yang berbasis standar open source yang dapat digunakan untuk membangun aplikasi-aplikasi mobile. Aplikasi yang dikembangkan dengan PhoneGap dapat dijalankan secara lintas *platform*. Hal ini dikarenakan development framework yang digunakan menggunakan bahasa scripting HTML, CSS, JavaScript.

XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL database, dan penterjemah bahasa yang ditulis dengan bahasa pemrograman PHP.

Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. XAMPP dapat compatible dengan 4 OS lainnya, makanya di sebut XAMP. XAMPP Web server tersedia dalam *GNU General Public License* dan *Freeware*, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkannya dapat mendownload langsung dari website resminya. **Download XAMPP Linux, Windows, Mac OS X, dan Solaris (Download Sesuai Operating Sistem anda).**

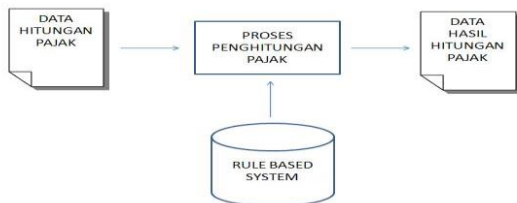
ANALISA DAN PERANCANGAN

Analisa Kebutuhan Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan dan hambatan yang terjadi serta memperbaikannya. Tahap analisis sistem dilakukan setelah tahap perencanaan sistem dan sebelum tahap desain sistem. Tahap analisis sistem merupakan tahap yang kritis dan sangat penting.

Deskripsi Sistem

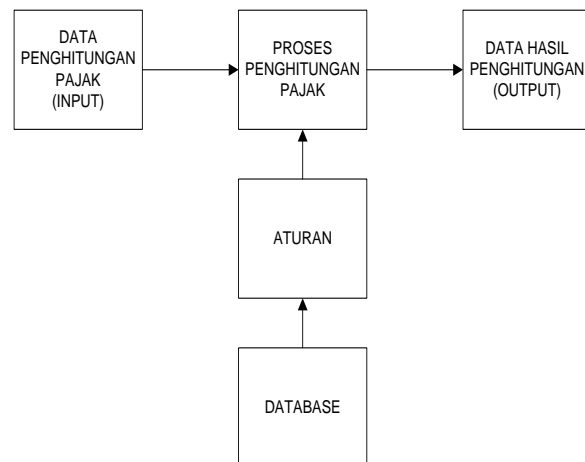
Deskripsi sistem adalah gambaran umum tentang sistem yang akan dikembangkan. Rancangan aplikasi penghitungan pajak penghasilan pasal 21 berbasis android merupakan perangkat lunak yang dapat digunakan untuk membantu menghitung pajak penghasilan yang diwujudkan dengan adanya dialog antara pengguna dengan sistem. Pada proses ini sistem akan menampilkan beberapa kriteria penghitungan untuk menentukan dan mengetahui besar pajak yang dibebankan.



Gambar 3.1 Garis Besar Rancangan Aplikasi

Desain Sistem

Pada desain sistem dilakukan pembuatan block diagram dimaksudkan untuk mengetahui lingkup proses dari rancangan aplikasi yang dibahas. Pada block diagram ini, dapat dilihat bahwa inputan berupa data perhitungan pajak yang kemudian akan diproses melalui basis aturan penghitungan pajak penghasilan untuk mengetahui besar pajak yang dibebankan.

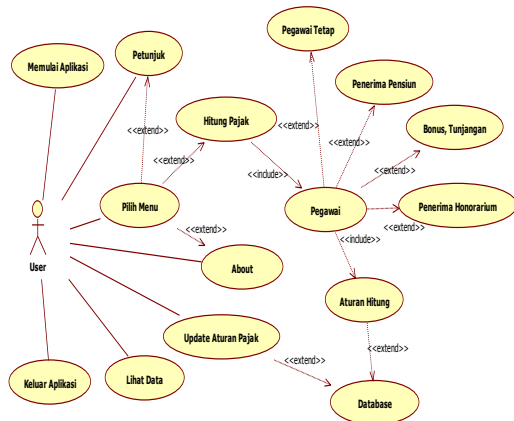


Gambar 3.2 Bisnis Proses

Keterangan :

Pada gambar 3.2 merupakan ruang lingkup proses dimana langkah awal yaitu memasukkan inputan yang berupa data perhitungan pajak yang kemudian akan diproses. Di dalam proses terdapat aturan penghitungan pajak penghasilan untuk menghitung dari input yang telah dimasukkan sehingga akan mendapatkan hasil berupa data hasil penghitungan pajak penghasilan yang dibebankan.

Usecase Diagram Aplikasi Penghitungan Pph 21



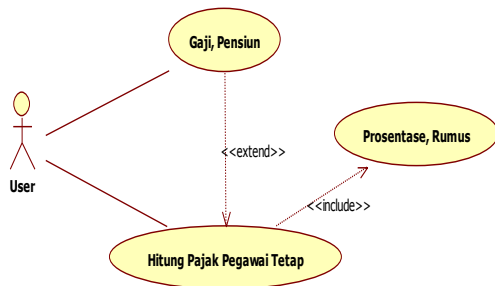
Gambar 3.3 Usecase Diagram Aplikasi Penghitungan Pph 21

Keterangan :

Pada gambar 3.3 merupakan *Usecase Diagram*, dimana *User* dapat menghitung, melihat data hasil penghitungan, melihat petunjuk penggunaan, mengupdate data tersebut. Dalam penghitungan pajak ada beberapa tipe penghitungan yang berbeda yaitu pegawai tetap, penerima pensiun, bonus/tunjangan, penerima honorarium, komisi penjaja barang dagangan, pendidikan pelatihan magang dan penghasilan upah harian.

Usecase Diagram Hitung Pajak Pegawai Tetap

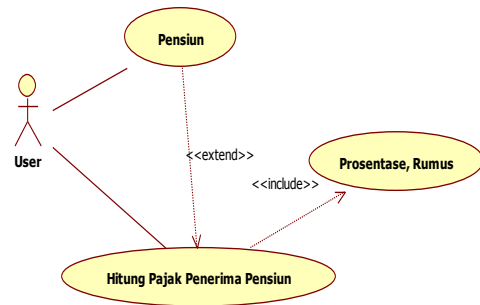
Usecase Diagram pada gambar berikut ini menjelaskan penghitungan pajak pegawai tetap yang dilakukan oleh user.



Gambar 3.4 Usecase Diagram Hitung Pajak Pegawai Tetap

Usecase Diagram Hitung Pajak Penerima Pensiun

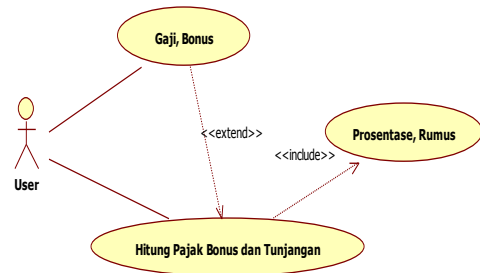
Usecase Diagram pada gambar berikut ini menjelaskan penghitungan pajak penerima pensiun yang dilakukan oleh user.



Gambar 3.5 Usecase Diagram Hitung Pajak Penerima Pensiun

Usecase Diagram Hitung Pajak Bonus dan Tunjangan

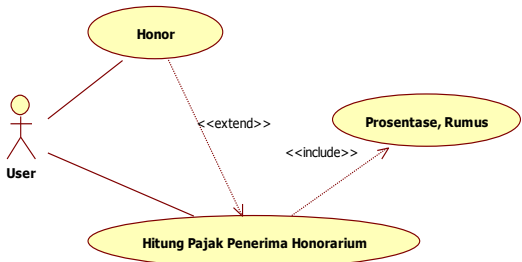
Usecase Diagram pada gambar berikut ini menjelaskan penghitungan pajak bonus dan tunjangan yang dilakukan oleh user.



Gambar 3.6 Usecase Diagram Hitung Pajak Bonus dan Tunjangan

Usecase Diagram Hitung Pajak Penerima Honorarium

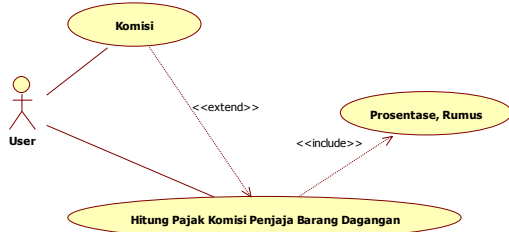
Usecase Diagram pada gambar berikut ini menjelaskan penghitungan pajak penerima honorarium yang dilakukan user.



Gambar 3.7 Usecase Diagram Hitung Pajak Penerima Honorarium

Usecase Diagram Hitung Pajak Komisi Penjaja Barang Dagangan

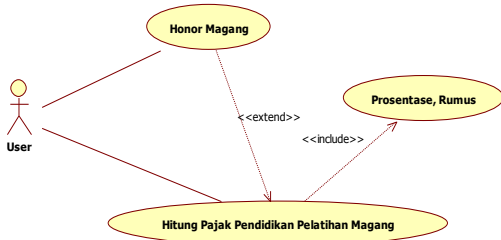
Usecase Diagram pada gambar berikut ini menjelaskan penghitungan pajak komisi penjaja barang dagangan yang dilakukan oleh user.



Gambar 3.8 Usecase Diagram Hitung Pajak Komisi Penjaja Barang Dagangan

Usecase Diagram Hitung Pajak Pendidikan Pelatihan Magang

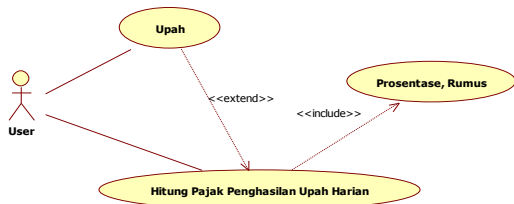
Usecase Diagram pada gambar berikut ini menjelaskan penghitungan pajak pendidikan pelatihan magang yang dilakukan oleh user.



Gambar 3.9 Usecase Diagram Hitung Pajak Pendidikan Pelatihan Magang

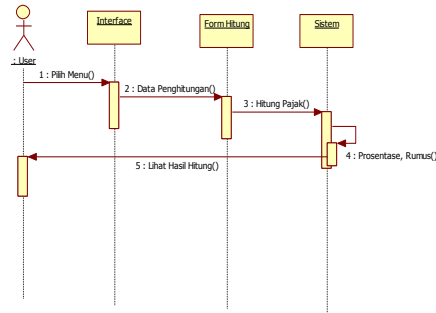
Usecase Diagram Hitung Pajak Penghasilan Upah Harian

Usecase Diagram pada gambar berikut ini menjelaskan penghitungan pajak penghasilan upah harian yang dilakukan user.



Gambar 3.10 Usecase Diagram Hitung Pajak Penghasilan Upah Harian

Sequence Diagram Aplikasi Penghitungan Pph 21



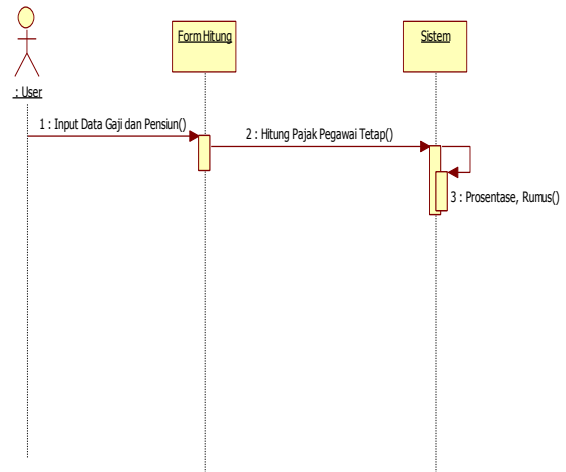
Gambar 3.11 Sequence Diagram Aplikasi Penghitungan Pph 21

Keterangan :

Pada gambar 3.11 merupakan *Sequence Diagram* dimana menampilkan alur proses berjalannya aplikasi yaitu *User* memilih menu penghitungan pajak dan menginputkan beberapa kriteria yang dibutuhkan untuk proses penghitungan, setelah itu sistem akan menghitung pajak penghasilan yang dipilih pada menu. Hasil penghitungan pajak akan tampil setelah proses penghitungan selesai. Kemudian sistem akan menyimpan data tersebut ke database.

Sequence Diagram Hitung Pajak Pegawai Tetap

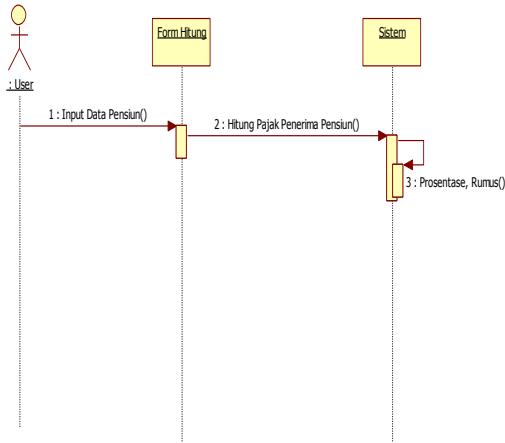
Sequence Diagram pada gambar berikut ini menjelaskan penghitungan pajak pegawai tetap yang dilakukan oleh user.



Gambar 3.12 Sequence Diagram Hitung Pajak Pegawai Tetap

Sequence Diagram Hitung Pajak Penerima Pensiun

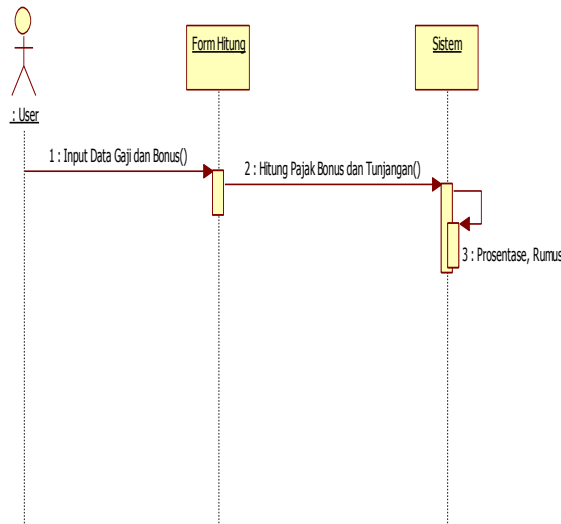
Sequence Diagram pada gambar berikut ini menjelaskan penghitungan pajak penerima pensiun yang dilakukan oleh user.



Gambar 3.13 Sequence Diagram Hitung Pajak Penerima Pensiun

Sequence Diagram Hitung Pajak Bonus dan Tunjangan

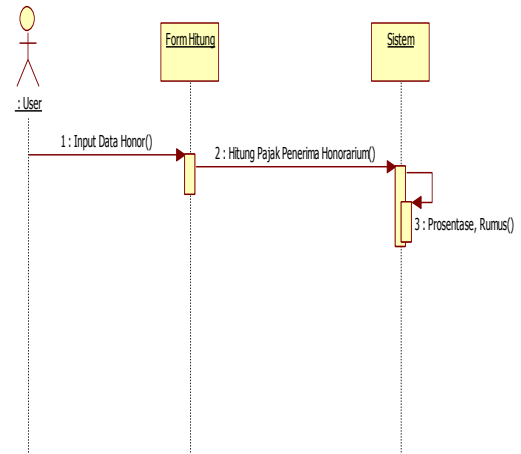
Sequence Diagram pada gambar berikut ini menjelaskan penghitungan pajak bonus dan tunjangan yang dilakukan oleh user.



Gambar 3.14 Sequence Diagram Hitung Pajak Bonus dan Tunjangan

Sequence Diagram Hitung Pajak Penerima Honorarium

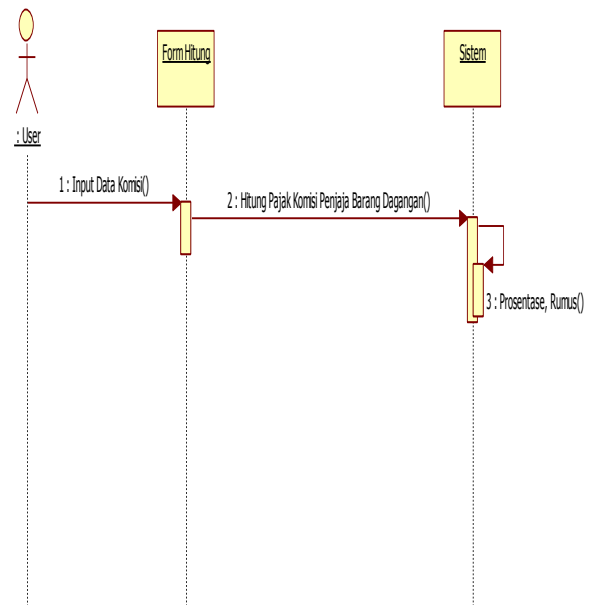
Sequence Diagram pada gambar berikut ini menjelaskan penghitungan pajak penerima honorarium yang dilakukan oleh user.



Gambar 3.15 Sequence Diagram Hitung Pajak Penerima Honorarium

Sequence Diagram Hitung Pajak Komisi Penjaja Barang Dagangan

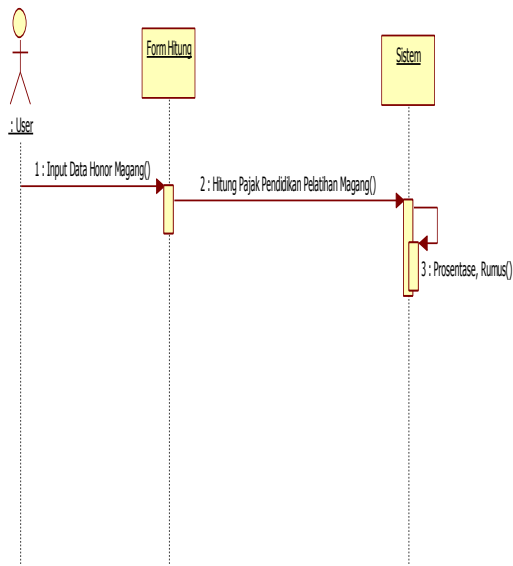
Sequence Diagram pada gambar berikut ini menjelaskan penghitungan pajak komisi penjaja barang dagangan yang dilakukan oleh user.



Gambar 3.16 Sequence Diagram Hitung Pajak Komisi Penjaja Barang Dagang

Sequence Diagram Hitung Pajak Pendidikan Pelatihan Magang

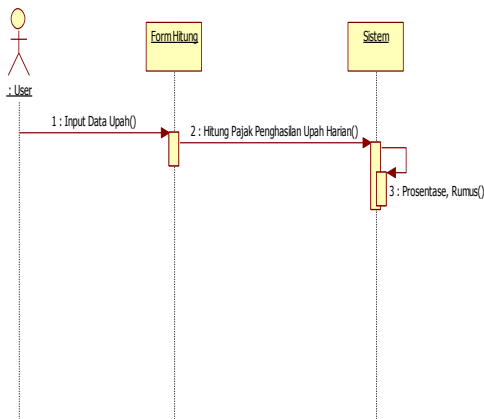
Sequence Diagram pada gambar berikut ini menjelaskan penghitungan pajak pendidikan pelatihan magang yang dilakukan oleh user.



Gambar 3.17 Sequence Diagram Hitung Pendidikan Pelatihan Magang

Sequence Diagram Hitung Pajak Penghasilan Upah Harian

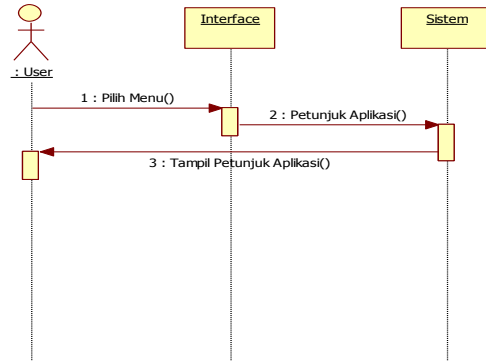
Sequence Diagram pada gambar berikut ini menjelaskan penghitungan pajak penghasilan upah harian yang dilakukan oleh user.



Gambar 3.18 Sequence Diagram Hitung Pajak Penghasilan Upah Harian

Sequence Diagram Petunjuk Penggunaan Aplikasi

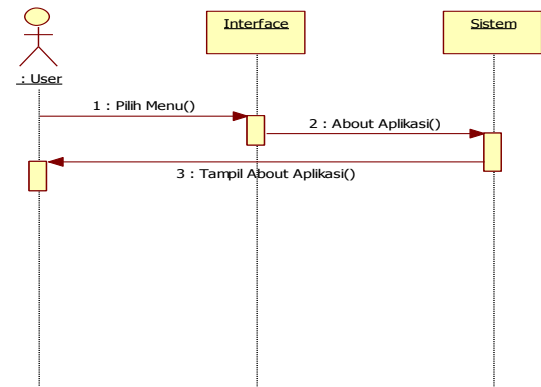
Sequence Diagram pada gambar berikut ini menjelaskan petunjuk penggunaan aplikasi pajak penghasilan yang dilakukan oleh user.



Gambar 3.19 Sequence Diagram Petunjuk Penggunaan Aplikasi

Sequence Diagram About/Tentang Aplikasi

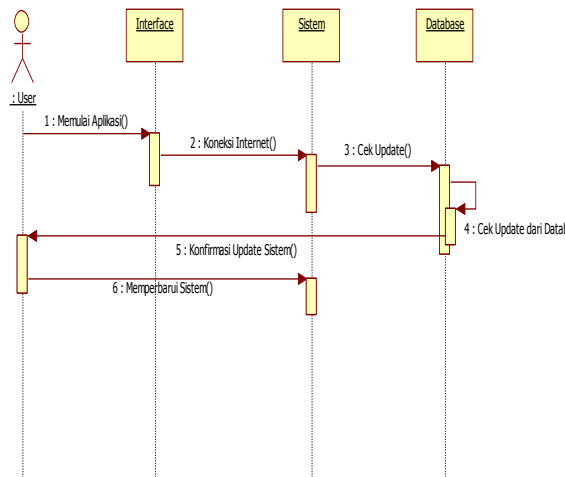
Sequence Diagram pada gambar berikut ini menjelaskan petunjuk penggunaan aplikasi pajak penghasilan yang dilakukan oleh user.



Gambar 3.20 Sequence Diagram About/Tentang Aplikasi

Sequence Diagram Update Sistem

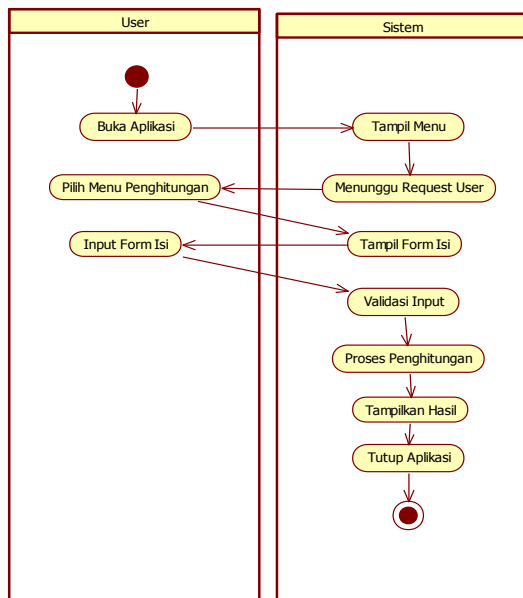
Sequence Diagram pada gambar berikut ini menjelaskan update sistem aplikasi pajak penghasilan yang dilakukan oleh user.



Gambar 3.21 Sequence Diagram Update Sistem

Activity Diagram Aplikasi Penghitungan Pph 21
Activity Diagram Hitung Pajak Penghasilan

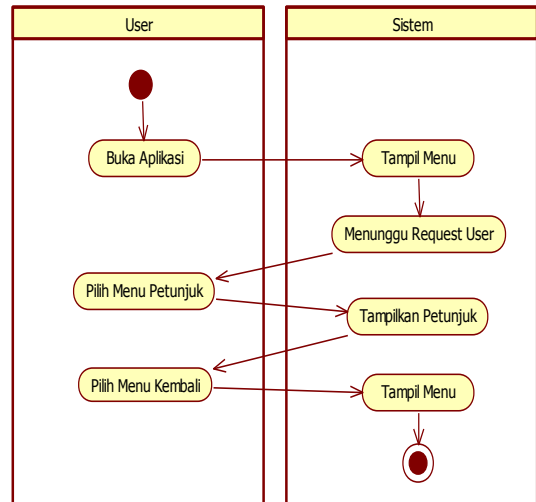
Menu Penghitungan merupakan inti dari aplikasi ini, yaitu tempat penghitungan PPh pasal 21 di proses. Di dalam menu Penghitungan terdapat form Isian yang menampung input data dari user, kemudian di cek kebenaran input dari user, setelah benar akan di hitung dan ditampilkan hasilnya di form Hasil. Activity diagram dari menu Penghitungan dapat dilihat dalam gambar berikut ini.



Gambar 3.22 Activity Diagram Hitung Pajak Penghasilan

Activity Diagram Petunjuk Penggunaan Aplikasi

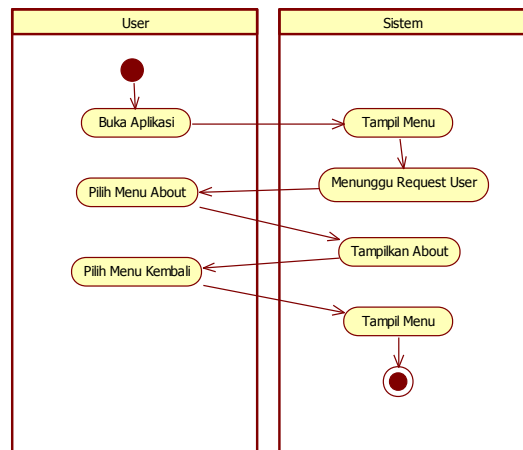
Pada Menu Petunjuk terdapat form yang menampilkan tentang cara menggunakan aplikasi. Berikut merupakan activity diagram untuk pengaksesan menu Petunjuk



Gambar 3.23 Activity Diagram Petunjuk Penggunaan Aplikasi

Activity Diagram About/Tentang Aplikasi

Menu petunjuk menyimpan informasi tentang aplikasi dan pembuat aplikasi yang akan di tampilkan kepada user. Activity diagram menu About terlihat dalam gambar berikut

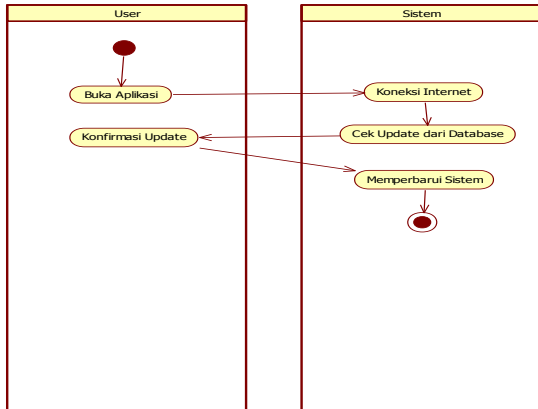


Gambar 3.24 Activity Diagram About/Tentang Aplikasi

Activity Diagram Update Sistem

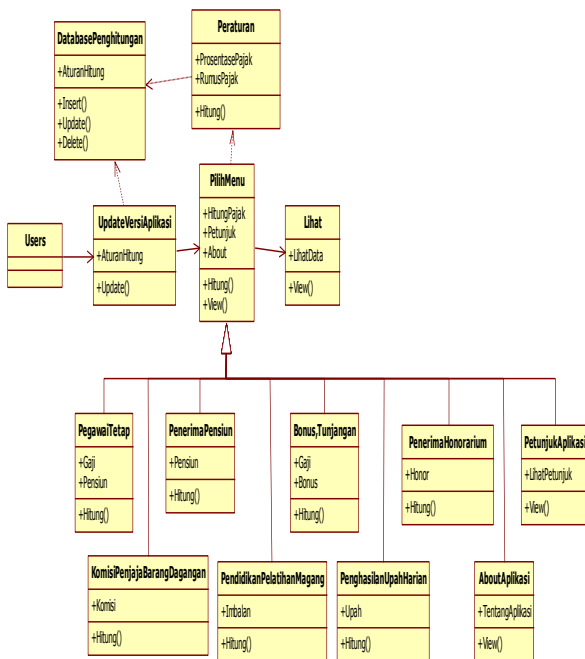
Setelah aplikasi dijalankan, sistem akan menanyakan apakah aplikasi ingin dijalankan secara offline atau online. Jika user memilih offline maka sistem tidak akan mengecek untuk update penghitungan yang baru jika tersedia, sebaliknya jika user memilih online maka sistem akan

mengecek update penghitungan yang baru jika tersedia, kemudian akan muncul konfirmasi update sistem dan sistem akan diperbarui. Activity diagram update sistem terlihat dalam gambar berikut



Gambar 3.25 Activity Diagram Update Sistem

Class Diagram Aplikasi Penghitungan Pph 21



Gambar 3.26 Class Diagram Aplikasi Penghitungan Pph 21

Keterangan :
 Pada gambar 3.26 merupakan *Class Diagram* dimana *User* memilih menu penghitungan terlebih dahulu sebelum melakukan penghitungan pajak pasal 21, menu tersebut diantaranya yaitu:

1. Pegawai Tetap, penghitungan ini membutuhkan data gaji dan pensiun sebagai kriteria untuk mengetahui besar pajak yang harus dibayar.
2. Penerima Pensiun, penghitungan ini hanya membutuhkan data pensiun sebagai kriteria untuk mengetahui besar pajak yang harus dibayar.
3. Bonus/Tunjangan, penghitungan ini membutuhkan data gaji dan bonus sebagai kriteria untuk mengetahui besar pajak yang harus dibayar.
4. Penerima Honorarium, penghitungan ini hanya membutuhkan data honor sebagai kriteria untuk mengetahui besar pajak yang harus dibayar.
5. Komisi Penjaja Barang Dagangan, penghitungan ini hanya membutuhkan data komisi sebagai kriteria untuk mengetahui besar pajak yang harus dibayar.
6. Pendidikan Pelatihan Magang, penghitungan ini hanya membutuhkan data imbalan sebagai kriteria untuk mengetahui besar pajak yang harus dibayar.
7. Penghasilan Upah Harian, penghitungan ini hanya membutuhkan data upah sebagai kriteria untuk mengetahui besar pajak yang harus dibayar.
8. Setelah melakukan proses penghitungan pajak pasal 21 sistem akan menampilkan hasil penghitungan pajak dan sistem akan menyimpan data hasil penghitungan kedalam database.

IMPLEMENTASI PERANGKAT LUNAK

Uji Coba Sistem

Pada sistem ini dihasilkan sebuah skenario pengujian, sehingga dapat menjelaskan urutan sistem yang telah dirancang. Beberapa skenario sistem pakar penentuan bagian harta *waris* meliputi:

Uji Coba Hitung PPH Pegawai Tetap

Pada Halaman Hitung PPH Pegawai Tetap, *user* diminta menginputkan jumlah gaji, pensiun dan memilih status penghasilan tidak kena pajak (ptkp) yang dimiliki.

The screenshot shows the 'Hitung Pajak' page for 'PPH Pegawai Tetap'. The user has entered the year 2012, a salary of Rp 20,000,000, and a pension of Rp 50,000. The status is set to 'K/0'. The calculated values are: Netto Bulan (Rp 18,500,000), Netto Tahun (Rp 222,000,000), Jumlah PTKP (Rp 171,600,000), PKP Setahun (Rp 50,400,000), PPH Setahun (Rp 252,000), and PPH Sebulan (Rp 21,000). A 'Kosongkan' button is visible at the bottom right.

Gambar 6.1 Uji Coba Hitung PPH Pegawai Tetap

Uji Coba Hitung PPH Penerima Pensiun

Pada Halaman Hitung PPH Penerima Pensiun, *user* diminta menginputkan pensiun dan memilih status penghasilan tidak kena pajak (ptkp) yang dimiliki.

The screenshot shows the 'Hitung Pajak' page for 'PPH Penerima Pensiun'. The user has entered the year 2012, a pension of Rp 3,000,000, and a status of 'K/1'. The calculated values are: Netto Bulan (Rp 2,850,000), Netto Tahun (Rp 34,200,000), Jumlah PTKP (Rp 184,800,000), PKP Setahun (Rp 157,200,000), PPH Setahun (Rp 786,000), and PPH Sebulan (Rp 65,500). A 'Kosongkan' button is visible at the bottom right.

Gambar 6.2 Uji Coba Hitung PPH Penerima Pensiun

Uji Coba Hitung PPH Bonus dan Tunjangan

Pada Halaman Hitung PPH Bonus dan Tunjangan, *user* diminta menginputkan jumlah gaji, tunjangan, pensiun dan memilih status penghasilan tidak kena pajak (ptkp) yang dimiliki.

The screenshot shows the 'Hitung Pajak' page for 'PPH Bonus & Tunjangan'. The user has entered the year 2012, a salary of Rp 25,000,000, and a bonus of Rp 1,000,000. The pension is Rp 50,000 and the status is 'K/0'. The calculated values are: Jumlah PTKP (Rp 171,600,000) and PPH Sebulan (Rp 47,500). A 'Kosongkan' button is visible at the bottom right.

Gambar 6.3 Uji Coba Hitung PPH Bonus dan Tunjangan

Uji Coba Hitung PPH Penerima Honorarium

Pada Halaman Hitung PPH Penerima Honorarium, *user* hanya diminta menginputkan jumlah honor.

The screenshot shows the 'Hitung Pajak' page for 'PPH Penerima Honorarium'. The user has entered a honorarium of Rp 15,000,000. The calculated PPH is Rp 37,500. A 'Kosongkan' button is visible at the bottom right.

Gambar 6.4 Uji Coba Hitung PPH Penerima Honorarium

Uji Coba Hitung PPH Komisi Penjaja Barang Dagangan

Pada Halaman Hitung PPH Komisi Penjaja Barang Dagangan, *user* hanya diminta menginputkan jumlah komisi.

The screenshot shows the 'Hitung Pajak' page for 'PPH Komisi Penjaja Barang Dagangan'. The user has entered the year 2012 and a commission of Rp 4,000,000. The calculated PPH is Rp 34,000. A 'Kosongkan' button is visible at the bottom right.

Gambar 6.5 Uji Coba Hitung PPH Komisi Penjaja Barang Dagangan

Uji Coba Hitung PPH Pendidikan Pelatihan Magang

Pada Halaman Hitung PPH Pendidikan Pelatihan Magang, *user* hanya diminta menginputkan jumlah imbalan.

Gambar 6.6 Uji Coba Hitung PPH Pendidikan Pelatihan Magang

Uji Coba Hitung PPH Penghasilan Upah Harian

Pada Halaman Hitung PPH Penghasilan Upah Harian, *user* diminta menginputkan jumlah upah dan memilih apakah *user* memiliki npwp atau tidak.

Gambar 6.7 Uji Coba Hitung PPH Penghasilan Upah Harian

Hasil Analisa dan Pengujian Sistem

Analisa dilakukan terhadap setiap menu penghitungan pajak penghasilan mulai dari inputan sebagai modal penghitungan dan cara penghitungan pajak penghasilan pada aplikasi mobile berbasis android.

Analisa Hitung PPH Pegawai Tetap

Dalam Penghitungan PPH Pegawai Tetap pada gambar 6.1.1 ini, *user* memiliki gaji Rp 2.000.000, pensiun sebesar Rp 50.000 dan status penghasilan tidak kena pajak (ptkp) adalah K/0 (kawin dan tidak memiliki tanggungan) sehingga setelah melalui proses penghitungan menurut penghitungan yang berlaku menghasilkan pph perbulan sebesar Rp 21.000.

Analisa Hitung PPH Penerima Pensiun

Dalam Penghitungan PPH Penerima Pensiun pada gambar 6.1.2 ini, *user* memiliki gaji Rp 3.000.000 dan status penghasilan tidak kena pajak (ptkp) adalah K/1 (kawin dan memiliki 1 tanggungan) sehingga setelah melalui proses penghitungan menurut penghitungan yang berlaku menghasilkan pph perbulan sebesar Rp 65.500.

Analisa Hitung PPH Bonus Dan Tunjangan

Dalam Penghitungan PPH Pegawai Bonus dan Tunjangan pada gambar 6.1.3 ini, *user* memiliki gaji Rp 2.500.000, bonus dan tunjangan Rp 1.000.000, pensiun sebesar Rp 50.000 dan status penghasilan tidak kena pajak (ptkp) adalah K/0 (kawin dan tidak memiliki tanggungan) sehingga setelah melalui proses penghitungan menurut penghitungan yang berlaku menghasilkan pph perbulan sebesar Rp 47.500.

Analisa Hitung PPH Penerima Honorarium

Dalam Penghitungan PPH Penerima Honorarium pada gambar 6.1.4 ini, *user* memiliki honor Rp 1.500.000 sehingga setelah melalui proses penghitungan menurut penghitungan yang berlaku menghasilkan pph perbulan sebesar Rp 37.500.

Analisa Hitung PPH Komisi Penjaja Barang Dagangan

Dalam Penghitungan PPH Komisi Penjaja Barang Dagangan pada gambar 6.1.5 ini, *user* memiliki komisi Rp 4.000.000 sehingga setelah melalui proses penghitungan menurut penghitungan yang berlaku menghasilkan pph perbulan sebesar Rp 34.000.

Analisa Hitung PPH Pendidikan Pelatihan Magang

Dalam Penghitungan PPH Pendidikan Pelatihan Magang pada gambar 6.1.6 ini, *user* memiliki imbalan Rp 1.500.000 sehingga setelah melalui proses penghitungan menurut penghitungan yang berlaku menghasilkan pph perbulan sebesar Rp 175.000.

Analisa Hitung PPH Penghasilan Upah Harian

1. Dalam Penghitungan PPH Pegawai Tetap pada gambar 6.1.7 ini, *user* memiliki upah Rp 200.000 dan tidak memiliki npwp sehingga setelah melalui proses penghitungan menurut penghitungan yang berlaku menghasilkan pph perbulan sebesar Rp 3000.

PENUTUP

Kesimpulan

Perhitungan Pajak Penghasilan Pph21 menggunakan aplikasi ini membantu pegawai atau karyawan yang ingin menghitung pajak tanpa integrasi dengan gaji dan aplikasi ini merupakan aplikasi yang mudah digunakan dimana saja dan kapan saja. Semua orang yang memiliki android bisa menggunakan aplikasi ini. Penerapannya cukup mudah dimengerti dengan ada menu petunjuk yang menuntun *user* dalam melakukan penghitungan pajak.

Selain itu terdapat menu update yang selalu membuat aplikasi selalu memiliki fitur penghitungan pajak terbaru.

Saran

Dalam pengembangan aplikasi Perhitungan Pajak Penghasilan (PPh21) disarankan untuk:

1. Ditambahkan database di dalam android agar memudahkan pengguna dalam menyimpan data hasil penghitungan pajak penghasilan.
2. Hasil penghitungan pajak penghasilan lebih dirinci agar pengguna dapat melihat perincian penghitungan, misalnya PTKP dan prosentase pph21.

DAFTAR PUSTAKA

- Gunadi. 2002. *Ketentuan Dasar Pajak Penghasilan, Edisi Pertama*. Jakarta: PT Salemba Emban Patria.
- Ivan Michael Siregar. 2011. *Membongkar source code berbagai Aplikasi Android*. Yogyakarta: Gava Media.
- Mardiasmo. 2003. *Perpajakan, Edisi Revisi*. Yogyakarta: Andi Offset.
- Marom, Chairul. 2002. *Akuntansi Pajak Penghasilan*. Jakarta: Pustaka Damar.
- Naci Dai, Lawrence Mandel, dan Arthur Ryman. 2007. *Eclipse Web Tools Platform: Developing Java Web Applications*.
- Nugroho. Adi. 2008. *Pemrograman Java Menggunakan IDE Eclipse*. Jakarta: Andi Publisher.

<http://bayduaenam.blogspot.com/2011/06/eclipse.html> *Masih Belajar Eclipse* Diakses tanggal 27 Maret 2013.

<http://developer.android.com/guide/index.html> / *Android SDK Developer Guide*, Diakses tanggal 24 Maret 2013.

http://opensource.telkomspeedy.com/wiki/index.php/Eclipse:_ADT_plugin_untuk_Eclipse Diakses Tanggal 2 April 2013.

{Halaman sengaja dikosongkan}

{Jurnal **Melek IT**}

(7)

PERANCANGAN PERMAINAN BERBASIS ANDROID SEBAGAI MEDIA BELAJAR UNTUK ANAK USIA DINI UMUR 2-4 TAHUN MENGGUNAKAN *AUGMENTED REALITY*

Anang Kukuh Adisusilo, S.T., M.T.

Program Studi Teknik Informatika, Fakultas Teknik
Universitas Wijaya Kusuma Surabaya
Jalan Dukuh Kupang XXV/54 Surabaya 60225
Jawa Timur, Indonesia
E-mail: anang@anang65.web.id

ABSTRAK

Permainan moderen sebagai salah satu media belajar mulai digunakan dalam penyelenggaraan proses belajar dan mengajar terutama ditingkat *playgroup* atau anak usia dini, usia antara 2-4 tahun. Pada usia tersebut kecenderungan bermain dapat digunakan sebagai bentuk motifasi dalam bermain sambil belajar. Keadaan ini mendorong para peneliti dan industri permainan terus mengembangkan berbagai permainan yang menarik dan tetap sesuai dengan aturan dalam penyelenggaraan pendidikan. Penggunaan alat bantu teknologi yang berkembang sangat mendukung dalam terlaksananya cara belajar melalui bentuk permainan, diantaranya munculnya berbagai perangkat yang berbasis *android*. Dengan peralatan berbasis android yang bersifat bergerak (*mobile*) dan harga sudah terjangkau oleh kalangan masyarakat secara umum, dapat mempermudah implementasi permainan. Keanekaragaman bentuk permainan dan peralatan untuk menjalankannya dengan jenis konten permainan yang sama, merupakan salah satu cara untuk meningkatkan daya tarik permainan sehingga tidak jenuh dalam bermain. Salah satu pemicu kejenuhan dalam bermain karena tingkat imersive dari permainan yang kurang, sehingga pengguna tidak merasa masuk lebih mendalam dalam alur permainan. Dalam penelitian ini akan dirancang suatu bentuk permainan bagi anak usia dini dengan meningkatkan nilai immersifitas permainan, sebagai contoh mengangkat konten permainan pengenalan huruf dan permainan pengenalan jenis-jenis buah. Teknologi yang digunakan adalah menggunakan AR (*Augmented Reality*) dengan model grafis tiga dimensi (3D). Dari hasil percobaan dengan menggunakan prototipe dihasilkan bentuk media belajar dengan immersifitas lebih tinggi dibandingkan memakai buku atau tanpa menggunakan AR, hal ini disebabkan grafis dan animasi 3D yang dihasilkan dari pengenalan *marker* (penanda) membuat pengguna seperti bersentuhan dengan objek sebenarnya.

Kata kunci: *augmented reality, permainan, android, pembelajaran, usia dini.*

I. PENDAHULUAN

Metode pembelajaran anak usia dini disekolah dan dirumah secara klasik adalah menggunakan buku-buku, seperti buku bergambar. Dengan kemajuan teknologi maka dikembangkan cara pembelajaran menggunakan media visual seperti audio dan video, sehingga anak-anak lebih tertarik dalam belajar. Metode belajar dengan media visual sudah lebih baik dibandingkan dengan metode klasik menggunakan buku-buku, tetapi belum cukup untuk bisa merangsang kecerdasan anak secara maksimal. Teknologi yang berkembang dewasa ini tidak menutup kemungkinan membuat pembelajaran lebih menarik lagi [1] dengan dimasukkan metode interaktif dalam belajar.

Teknologi yang interaktif dan banyak digunakan masyarakat dalam kesehariannya adalah teknologi permainan moderen (*game*). Bagi anak-anak yang masih senang bermain, untuk melakukan

permainan juga sudah memanfaatkan penggunaan teknologi. Permainan sudah menjadi hal pokok bagi anak-anak dalam aktifitas setiap harinya, dengan adanya teknologi *mobile* lebih mempermudah bagi anak-anak untuk bisa melakukan aktifitas permainan dimanapun tidak harus didepan *personal computer* (PC). Teknologi *mobile* dan *gadget* yang banyak digunakan adalah berbasis *android*. Kecenderungan tergantung pada permainan ini bisa dimanfaatkan untuk menjadikan permainan sekaligus sebagai sarana belajar.

Pembelajaran yang dimasukkan ke dalam game seperti pengenalan angka, cara berhitung, pengenalan warna, pengenalan huruf, pengenalan benda dan sebagainya sehingga anak akan merasa senang dengan sarana belajar sambil bermain [2]. Sudah banyak terdapat pilihan permainan, sebagian besar permainan menitik beratkan pada genre dan konten permainanannya bukan menitik beratkan pada kemampuan immersifitas sebuah permainan .

Sebuah permainan dengan menggunakan AR yang merupakan teknologi dengan konsep visual secara grafis di dunia nyata melalui sebuah kamera bisa menambah immersifitas permainan tersebut [3,4]. AR mentransformasi perangkat kamera ke dalam sesuatu yang digambarkan sebagai suatu cermin, sehingga pengguna dapat berinteraksi dan merasa seolah-olah berada dalam keadaan sebenarnya.

Dalam penelitian ini akan dirancang suatu bentuk permainan bagi anak usia dini dengan meningkatkan nilai immersifitas permainan, sebagai contoh mengangkat konten permainan pengenalan huruf dan permainan pengenalan jenis-jenis buah. Teknologi yang digunakan adalah menggunakan AR dengan model grafis 3D.

II. AUGMENTED REALITY

AR (*Augmented Reality*) merupakan teknologi yang menggabungkan dunia nyata dengan dunia maya. Teknologi *augmented reality* membutuhkan empat elemen utama yaitu untu *input*, *output/display*, *alat tracking* dan komputer sebagai alat proses. Alat *input* berupa *marker* yaitu seperti penanda, *output* dapat juga berupa perangkat keras seperti telepon, *tablet PC*, maupun alat-alat *mobile* lainnya, atau monitor/ *display*. Alat *tracking* dapat berupa kamera atau sensor lainnya, GPS, sensor nirkabel, kompas, dan lain-lain.

A. Input

Masukan dalam teknologi AR terdapat dua jenis, yaitu :

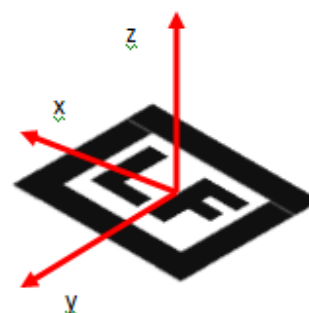
1. Marker

Marker sering disebut sebagai *marker based tracking* biasanya merupakan ilustrasi hitam dan putih persegi dengan batas hitam tebal dan latar belakang putih. Marker base tracking bekerja dengan 2 dimensi yang diberi barcode untuk mengkatifkan program AR, gambar.1. Komputer akan mengenali posisi dan orientasi marker dan menciptakan dunia virtual 3D yaitu titik (0,0,0) dan 3 sumbu yaitu X,Y,dan Z, gambar.2. *Marker based tracking* ini sudah lama dikembangkan sejak tahun 80an dan pada awal 90an mulai dikembangkan untuk penggunaan AR.

Marker based tracking memerlukan kamera *handphone* atau *webcam* yang cukup dengan intensitas cahaya. Keunggulannya adalah lebih mudah mengembangkan aplikasi karena untuk *trigger* hanya perlu memerlukan kamera.



Gambar.1 .Contoh Marker



Gambar.2 .Titik koordinat marker

2. Markerless

Metode *markerless* pengguna tidak perlu lagi menggunakan sebuah *marker* untuk menampilkan obyek-objek *virtual*. Seperti yang saat ini dikembangkan oleh perusahaan AR terbesar di dunia Total Immersion, mereka telah membuat berbagai macam teknik *Markerless Tracking* sebagai teknologi andalan mereka, seperti *Face Tracking*, *3D Object Tracking*, *Motion Tracking* dan *GPS Based Tracking*. Metode ini disebut MAR (*Markerless Augmented Reality*), contoh penggunaan seperti gambar.3, dengan adanya kursi muncul secara *virtual lego* yang duduk di kursi tersebut.



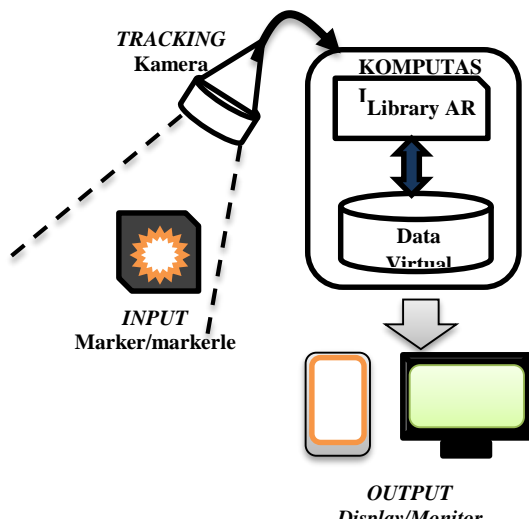
Gambar.3 .Contoh aplikasi MAR [5]

B. Output

Output dari AR merupakan objek dalam bentuk multimedia bisa berupa text, gambar dan juga video. Output tersebut ditampilkan dalam monitor atau *display* dari peralatan yang digunakan.

C. Tracking

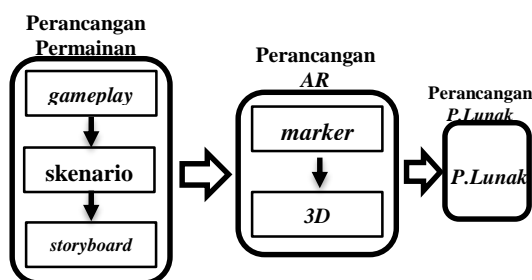
Tracking adalah peralatan yang digunakan untuk mengambil data dari penanda (*marker/markerless*) yang dibuat. Dari data hasil tracking dimasukkan sebagai inputan proses komputasi sehingga menghasilkan output yang diinginkan. Untuk lebih jelasnya seperti diagram gambar.4.



Gambar.4 .Diagram alir proses AR

III. PERANCANGAN SITEM

Permainan pada penelitian ini mengangkat konten permainan pengenalan huruf dan pengenalan buah. Secara umum tahapan penelitian dibedakan menjadi perancangan permainan, perancangan AR dan perancangan perangkat lunak kemudian dilakukan ujicoba dari hasil prototipe, seperti gambar.5.



Gambar.5 .Tahapan penelitian

A. Perancangan permainan

Perancangan permainan dibagi menjadi beberapa tahapan, yaitu :

1. Gameplay

Gameplay dalam permainan dengan menggunakan AR ini yaitu dengan menggunakan perangkat keras berbasis android. Kamera dari perangkat keras tersebut diarahkan ke *marker* dari huruf atau gambar buah. Dari *display* perangkat keras akan muncul visualisasi 3D sesuai dengan *marker* yang dikenali oleh sistem permainan. Jika ujung jari diletakkan pada ruang diantara *marker* dan kamera, muncul animasi dari objek 3D dan suara yang menjelaskan objek tersebut. Jika jari bisa menyentuh objek maka objek tersebut akan berubah warna . *Reward* yang diberikan ke pengguna adalah hitungan angka yang bertambah setiap objek dapat disentuh oleh ujung jari .

2. Skenario.

Perancangan skenario dalam permainan merupakan interaksi dari pemain dengan *virtual button* terhadap virtualisasi 3D yang dihasilkan, yaitu :

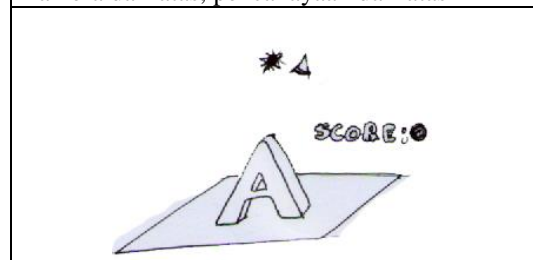
- Visualisasi 3D terlihat.
- Jari diletakkan diantara *marker* dan kamera, untuk menyentuh *virtual button*, kemudian muncul suara sesuai jenis objek, misalnya “buah apel”, “huruf A”.
- Untuk permainan pengenalan buah, animasi buah berputar atau berguling secara acak. Jika ujung jari mengenai buah, maka buah akan berubah warna dan score bertambah.
- Untuk permainan pengenalan huruf, animasi huruf berputar atau berguling secara acak. Jika ujung jari mengenai huruf, maka huruf akan berubah warna dan score bertambah.
- Permainan selesai setelah mendapatkan score 10, atau jari berhasil 10 kali menyentuh objek.
- Permainan akan trus berjalan walaupun *marker* dipindahkan.

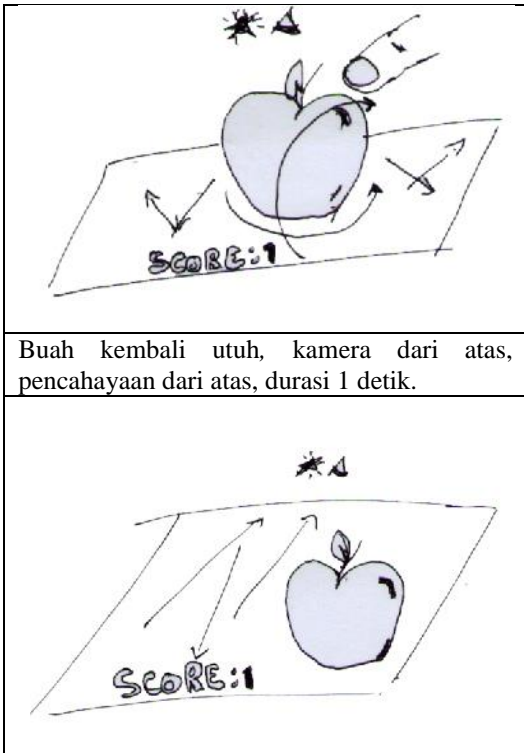
3. Storyboard.

Storyboard pada permainan dengan menggunakan AR untuk pengenalan huruf dan pengenalan buah adalah sebagai berikut:

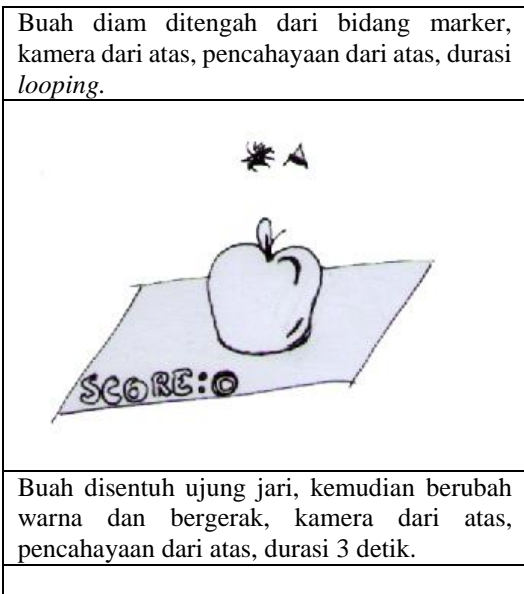
- Storyboard* untuk permainan pengenalan huruf.

Huruf diam ditengah dari bidang *marker*, kamera dari atas, pencahayaan dari atas





b. *Storyboard* untuk permainan pengenalan buah.



B. Perancangan AR

Pada perancangan AR ini dibagi menjadi beberapa bagian perancangan, yaitu perancangan marker dan perancangan objek 3D.

1. *Marker*

Marker diambilkan dari gambar-gambar yang berasal dari media belajar anak usia dini, seperti dalam bentuk buku bergambar atau kartu cerita bergambar. Beberapa contoh marker seperti gambar.12. dan gambar.13.



Gambar.12. contoh marker untuk pengenalan buah



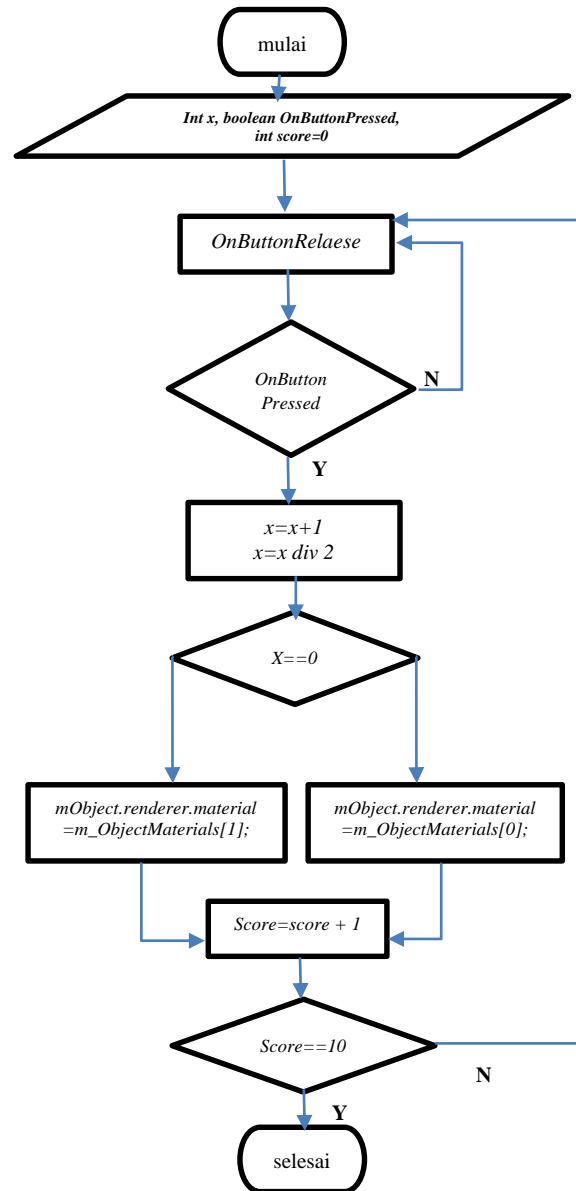
Gambar.13. Contoh marker untuk pengenalan huruf

2. Objek 3D.

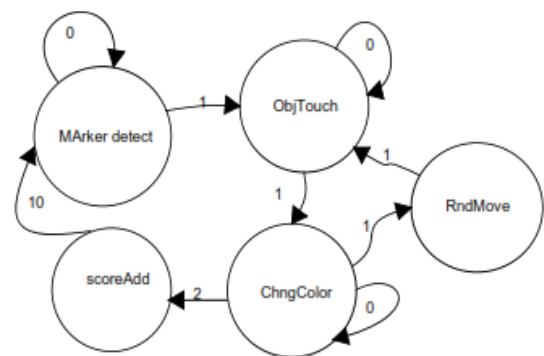
Objek tiga dimensi dibuat untuk visualisasi dari AR, dan dibuat menggunakan *tools modeling 3D* yaitu Blender [6,7,8]. Contoh objek yang dibuat pada penelitian ini adalah bentuk 3D dari buah Apel, buah Rambutan, huruf O dan huruf A.

C. Perancangan perangkat lunak

Dalam pembuatan perangkat lunak digunakan bahasa program C# dengan menggunakan *editor Mono* dan debugger *Unity game engine* [9,10]. Untuk mengenali marker digunakan *engine* dari *Qualcomm vuforia*, dan pada bagian *scoring* dan perubahan *action* perubahan warna objek dibuat perangkat lunak seperti *flowchart* pada gambar.14. Secara umum perancangan perangkat lunak permainan digambarkan seperti *Finite State Machine (FSM)* gambar.15.

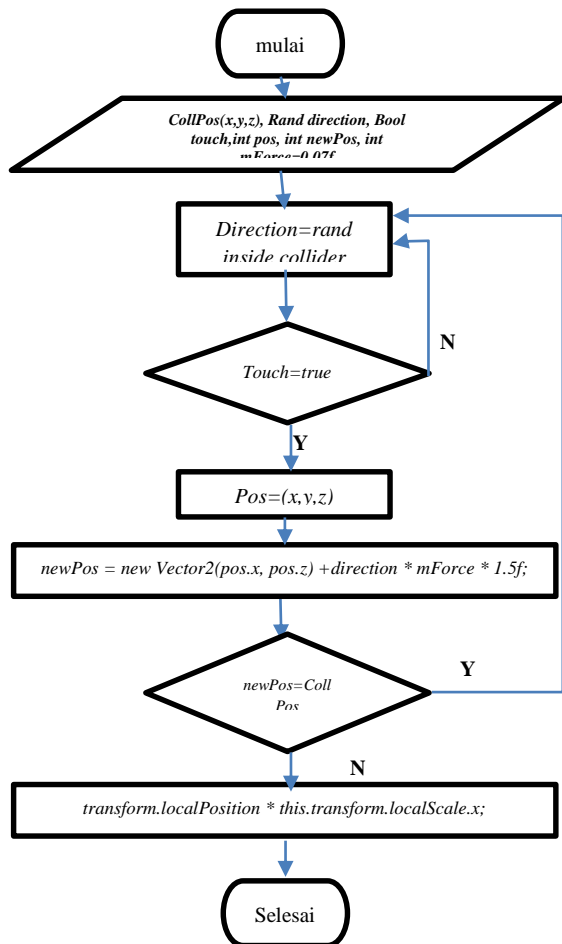


Gambar.14. Flowchart action perubahan warna dan scoring

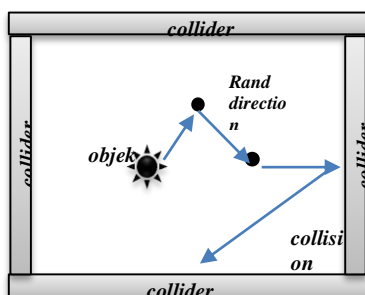


Gambar.15. FSM Game

Dalam pergerakan objek dibatasi dengan bentuk segi empat, sehingga objek tidak bisa keluar dari garis batas tersebut, pembatasan ini disebut sebuah *collider*. Objek bergerak secara random dan menggunakan fungsi *physics rigidbody game*. *Flowchart* dari gerakan objek seperti gambar.16. dan ilustrasi pembatas dari gerakan seperti gambar.17.



Gambar.16. *Flowchart* gerakan objek 3D



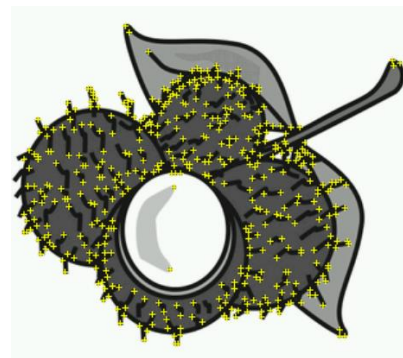
Gambar.17. Batas pergerakan objek 3D

IV. IMPLEMENTASI SISTEM

Implementasi diawali dengan pembuatan prototipe berdasarkan perancangan sistem yang dibuat. Dalam implementasi digunakan pada perangkat berbasis android.

A. Marker.

Dengan menggunakan *egine qualcomm vuforia* [11], didapatkan *patern marker* yang dimasukkan dalam *unity game engine*. Contoh hasil *patern marker* buah seperti gambar.18. dan gambar.19. Tanda bintang-bintik kecil seperti tanda plus (+) menunjukkan bagian yang diidentifikasi sebagai *patern marker* tersebut.

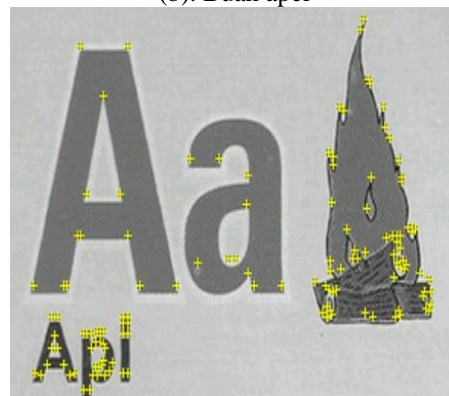


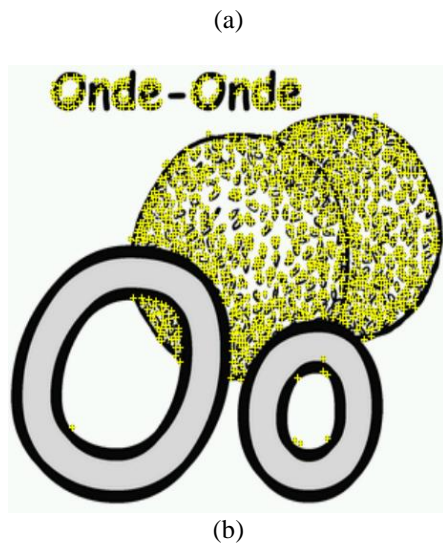
(a)



(b)

Gambar.18. *Patern marker*, (a). Buah rambutan, (b). Buah apel

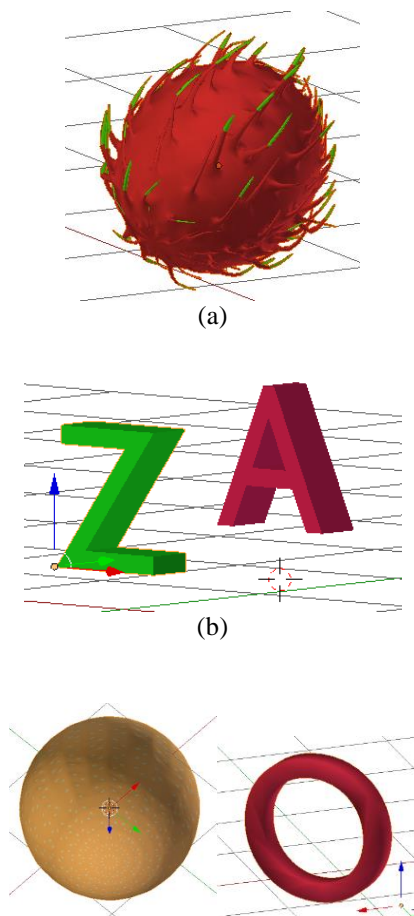




Gambar.19. Patern marker, (a). Huruf A-untuk Api,
(b). Huruf O- untuk Onde-onde

B. Model 3D.

Dengan menggunakan Blender dihasilkan model 3D, seperti contoh hasil gambar.20.



Gambar.20. Model 3D, (a). Rambutan,
(b). Huruf A dan Z. (c). Onde-Onde dan huruf O.

C. Perangkat Lunak.

Pada bagian implementasi perangkat lunak untuk proses perubahan warna dan *scoring* menggunakan bahasa program C# dengan *editor mono*.

```
public Material[] m_buahMaterials;
private GameObject mbuah;
private List<Material> mActiveMaterials;
private int score;
void ScoreColor () {
    mbuah =
    transform.Find("soccerball/soccerballmesh").gameObject;
    mbuah.renderer.material =m_buahMaterials[1];
    x = x + 1;
    if (buttonPress) {
        if (x % 2==0){
            mbuah.renderer.material =m_buahMaterials[0];
        }
        if (x % 2!=0){
            mbuah.renderer.material =m_buahMaterials[1];
        }
        score=score+1;
    }
}
```

Pada bagian implementasi perangkat lunak untuk proses gerakan dari objek menggunakan bahasa program C# dengan *editor mono*.

```
private GameObject mSoccerball;
private bool mIsRolling = false;
private float mTimeRolling = 0.4f;
private float mForce = 0.7f;

void moving() {
    Bounds targetBounds = this.collider.bounds;
    Rect targetRect = new Rect( -targetBounds.extents.x,
    -targetBounds.extents.z,
    targetBounds.size.x,
    targetBounds.size.z);
    Vector2 randomDir = new Vector2();
    for (int i = 0; i < 1; i++)
    {
        randomDir = Random.insideUnitCircle.normalized;
        Vector3 pos = mSoccerball.transform.localPosition *
        this.transform.localScale.x;
        Vector2 finalPos = new Vector2(pos.x, pos.z) +
        randomDir * mForce * 1.5f;
        if (targetRect.Contains(finalPos))
        {
            break;
        }
    }

    Vector3 kickDir = new Vector3(randomDir.x,
    0,randomDir.y).normalized;
    Vector3 torqueDir =
    Vector3.Cross(Vector3.up,kickDir).normalized;
    mSoccerball.rigidbody.AddForce(kickDir * mForce,
```

```
ForceMode.VelocityChange);
mBuah.rigidbody.AddTorque(torqueDir * mForce,
```

```
ForceMode.VelocityChange);
mIsRolling = true;
mTimeRolling = 0.0f;
```

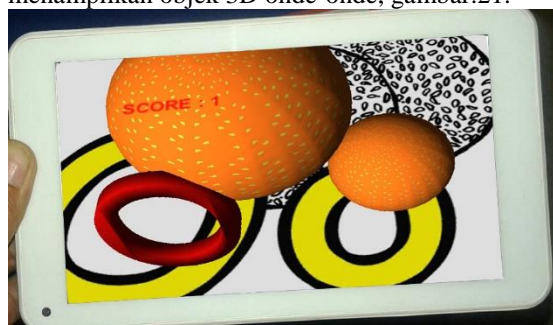
```
}
```

V. UJICOBA SISTEM

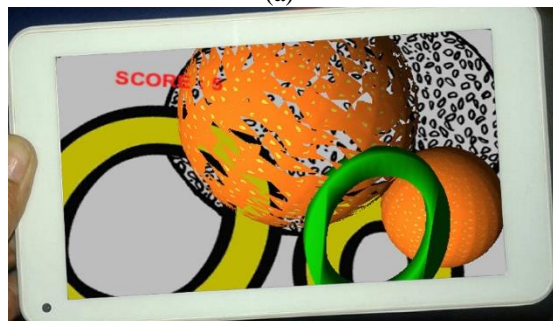
Untuk menganalisa sistem, prototipe yang dihasilkan di-build untuk dijalankan pada perangkat berbasis *android*, yaitu dengan file *.apk*.

A. Ujicoba permainan pengenalan huruf.

Contoh uji coba pada permainan pengenalan huruf, yaitu pada marker huruf O dengan menampilkan objek 3D onde-onde, gambar.21.



(a)



(b)

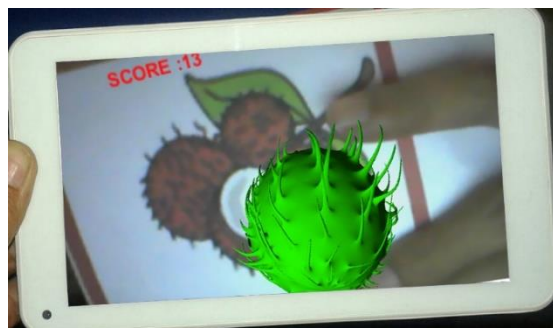
Gambar.21. Prototipe permainan pengenalan huruf, (a). Huruf O dengan objek onde-onde, (b).

Huruf O yang berubah warnanya dengan objek onde-onde

Gambar.21. menunjukkan prototipe permainan dengan menggunakan *augmented reality* untuk pengenalan huruf berhasil diimplementasikan sesuai perancangan game yang dibuat. Score berubah dan warna huruf berubah ketika tangan menyentuh objek 3D dari objek tersebut, selain itu huruf bergerak secara acak dengan batasan *collision* dalam area *marker*, gambar.17.

B. Ujicoba permainan pengenalan buah.

Sebagai contoh uji coba pengenalan huruf digunakan marker buah rambutan dan dihasilkan seperti gambar 22.



(a)



(b)

Gambar.22. Prototipe permainan pengenalan buah, (a). Buah rambutan berwarna hijau, (b). Buah rambutan berwarna merah

Gambar.22. menunjukkan prototipe permainan dengan menggunakan *augmented reality* untuk pengenalan buah berhasil diimplementasikan sesuai perancangan game yang dibuat. Score dan warna buah akan berubah ketika tangan menyentuh objek 3D dari buah tersebut, selain itu buah bergerak secara acak dengan batasan *collision* dalam area *marker*, gambar.17.

Prototipe dari permainan yang sudah dihasilkan dilakukan *survey*, dengan tujuan untuk mengetahui ketertarikan masyarakat terhadap bentuk permainan dengan *augmented reality*. *Survey* dilakukan terhadap 20 guru *playgroup* dan dihasilkan bahwa 90% tertarik media belajar menggunakan *augmented reality* dengan alasan bahwa seakan-akan menyentuh objek sesungguhnya, sedangkan 10% masih kurang tertarik dengan alasan faktor kesulitan jika anak kecil memakai alat berbasis *android*, misalnya tablet.

VI. KESIMPULAN

Dari penelitian yang dilakukan dapat diambil kesimpulan sebagai berikut :

1. Dengan perancangan yang dilakukan dan menggunakan engine marker *vuforia* dapat dihasilkan prototipe permainan sebagai media belajar pengenalan huruf dan buah untuk anak usia dini umur antara 2-4 tahun.
2. Immersifitas dari pengguna lebih tinggi dibandingkan dengan permainan tanpa menggunakan augmented reality hal ini terbukti dari ketertarikan pengguna mencapai 90%.
3. Dengan media belajar berupa permainan yang dilengkapi teknologi *augmented reality* dapat digunakan sekaligus sebagai media mengenalkan muatan lokal dari bangsa Indonesia seperti bentuk prototipe objek *Onde-onde*, yang merupakan makanan khas daerah Jawa.

VII. DAFTAR PUSTAKA

- [1] Hyungsung Park, "Relationship between Motivation and Student's Activity on Educational Game", *International Journal of Grid and Distributed Computing* Vol. 5, No. 1, March, 2012, Kyonggi University, Korea.
- [2] Arizonanataliya Heryanti, Galih Hermawan, *Game Shopping Time*, *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, Vol. 2, No. 1, Maret 2013, ISSN : 2089-9033.
- [3] Calleja G, "In-Game: From Immersion to Incorporation", *In-Game: From Immersion to Incorporation*. Cambridge, MA; London: MIT Press 2011.
- [4] Murray, H. J., "Immersion. In H. J. Murray (Ed.) *Hamlet on the Holodeck* ", Cambridge, Mass: MIT Press 1998 (pp. 97-125).
- [5] COMPORT, A. I. et al, "Real-time markerless tracking for augmented reality: The virtual visual servoing framework", *IEEE Transactions on Visualization and Computer Graphics*, IEEE Computer Society, Los Alamitos, CA, USA, v. 12, n. 4, p. 615–628, 2006. ISSN 1077-2626.
- [6] Blender modeling,"Documentation and user manual", <http://wiki.blender.org/> , akses pebruari 2014.
- [7] Lance Flavell, " *Beginning Blender: Open Source 3D Modeling, Animation, and Game Design* ", Copyright © 2010 by Lance Flavell, Apress, ISBN-13 (pbk): 978-1-4302-3126-4 ISBN-13 (electronic): 978-1-4302-3127-1.
- [8] Mythravarun Vepakomma, "Blender Compositing and Post Processing", Packt Publishing 2014, ISBN:978-1-78216-112-7
- [9] Unity game engine , "Tutorial Unity 3D", <http://unity3d.com> , diakses maret 2014.
- [11]Unity script ,"Forum scripting reference", <http://docs.unity3d.com>, diakses maret 2014.
- [10]Qualcomm Vuforia, "Documentation", <https://www.qualcomm.com/>, diakses maret 2014.

{Halaman sengaja dikosongkan}

{Jurnal **Melek IT**}

(6)

VIDEO COMPOSITING MENGGUNAKAN POISSON BLENDING

Saiful Yahya^{a1}, Mochamad Hariadi^{b2}, Ahmad Zaini^{b3}, and Eko Mulyanto Yuniarno^{b4}^aTeknik Elektro Bidang Keahlian Jaringan Cerdas Multimedia, Institut Teknologi Sepuluh Nopember^bMochamad Hariadi Teknik Multimedia dan Jaringan Institut Teknologi, Sepuluh Nopembere-mail: mochar@ee.its.ac.id², zaini@ee.its.ac.id³, ekomulyanto@ee.its.ac.id⁴

ABSTRAK

Penggunaan gradasi yang halus pada penggabungan dua video pada proses *video kompositing*. Video terdiri dari gabungan citra yang disusun secara teratur sehingga menimbulkan gerakan yang berkesinambungan. Proses penggabungan video pada dasarnya sama dengan penggabungan citra diam, tetapi untuk lebih memperhalus video diperlukan filter yang berbeda dengan filter yang digunakan pada citra diam. Penggunaan filter pada bidang 2D untuk memperhalus penggabungan antara 2 buah citra. Sedangkan pada video diperlukan filter 3D. Pada penelitian ini fokus pada *filter 3D poisson blending* untuk meningkatkan intensitas. *Poisson blending* berasal dari turunan kedua persamaan Laplacian. Untuk menerapkan persamaan laplacian pada video digunakan metode konvolusi untuk meng- gabungkan matriks dengan citra. Pengujian dilakukan terhadap video yang memiliki latar homogen, gelap dan terang, baik untuk video sumber dan video target. Dari hasil pengujian didapat nilai PSRN terbaik pada video sumber berlatar gelap yang digabungkan dengan video target berlatar gelap dengan nilai PSRN mencapai 60, 0658 pada frame pertama dan kedua. Pada frame pertama video sumber dan target dioperasikan terhadap frame pertama *matriks 3D poisson blending*. Nilai intensitas berdasarkan PSRN pada frame 2 dan seterusnya cenderung naik kualitasnya hingga lebih dari 10 desibel, sedangkan berdasarkan MSE nilainya turun hingga 0,1 sehingga disimpulkan intensitasnya cenderung lebih baik jika dibandingkan dengan frame pertama.

Kata Kunci : *3D poisson blending, intensitas, video kompositing, konvolusi*

I. PENDAHULUAN

Penggunaan latar belakang dalam proses pembuatan film dan dunia broadcast umum dilakukan untuk menghemat serta menambah tata artistic .

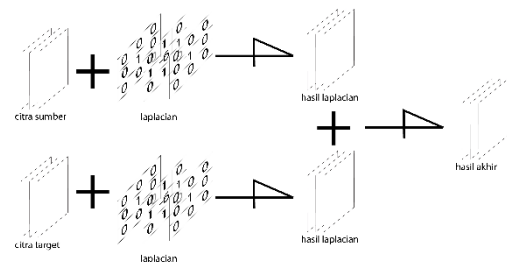
Penggunaan latar hijau atau biru digunakan untuk kemudian digantikan dengan citra bergerak atau citra diam. Penggunaan persamaan Poisson pada citra digital diperkenalkan oleh Perez pada tahun 2003 yang menyatakan menggunakan generik mesin interpolasi berbasis pada pemecahan Poisson persamaan [1]. Gradien domain blending memecahkan masalah tersebut dengan mentransfer gradien patch sumber gambar ke gambar target tetap menjaga batas blending mulus dan mengoreksi perbedaan pencahayaan. Hal ini dapat dilakukan dengan memecahkan persamaan Poisson, atau dengan interpolasi menggunakan Mean Value Coordinate (MVC).



Gambar 1: Penggunaan green screen pada proses syuting film (Foster, Jeff; 2010)

II. METODOLOGI PENELITIAN

Input dari system berupa dua jenis citra, yaitu citra sumber dan citra target. Citra sumber adalah citra yang mengalami proses pemotongan latar belakang dan citra target merupakan citra yang tidak mengalami pemotongan latar belakang serta sebagai tempat penggabungan dengan citra sumber.



Gambar 2: Diagram Penelitian

- 1) Pengubahan Video menjadi citra tunggal jamak. Pengubahan citra bergerak menjadi citra diam (tunggal) berprinsip bahwa video tersusun dari rangkaian citra diam yang ditampilkan secara berkesinambungan sehingga menimbulkan kesan citra bergerak.
- 2) Konversi Warna Citra target dan citra sumber diubah dari warna RGB menjadi Grayscale dengan persamaan (1). Hal ini dilakukan untuk menyederhanakan penghitungan.

$$S = \frac{r + g + b}{3} \tag{1}$$

- 3) Poisson blending. Poisson blending atau disebut juga operator laplace memiliki persamaan 2D seperti pada persamaan (2).

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y) \tag{2}$$

Jika dijadikan dalam bentuk matriks menjadi:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{3}$$

Dari matriks 2D di jadikan menjadi bentuk 3D matriks, sebagai berikut:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & -6 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4}$$

- 4) Konvolusi. Operasi konvolusi matriks poisson blending terhadap image:

Dimana α merupakan dummy variable of integration dan $g(x)$ adalah konvolusi kernel (dalam penelitian ini merupakan matriks 3D poisson). Konvolusi matriks dilakukan terhadap matriks T , $T - 1$ dan $T + 1$. Sebelum

matriks dioperasikan konvolusi, matriks diubah menjadi vektor agar dapat dikonvolusikan.

Konvolusi dilakukan pada masing-masing piksel citra sumber dan citra target terhadap matriks poisson. Hasil konvolusi citra sumber dan konvolusi citra target digabungkan untuk menghasilkan citra

akhir. pada saat konvolusi terjadinya perubahan

$$Z_{-\infty}^{\infty} f(x) * g(x) = f(x)g(x - \alpha) \tag{5}$$

ukuran video menjadi bertambah 2 piksel, 1 piksel bertambah disisi kanan dan 1 piksel bertambah disisi bawah citra, hal ini diabaikan untuk memudahkan penghitungan.

P SN R > 30, maka dapat dikategorikan bahwa performansi citra hasil bagus. Jika nilai P SN R > 50 maka dapat dikatakan bahwa persormansi citra hasil restorasi sempurna mendekati citra asli [2]. Untuk menentukan PSNR, terlebih dahulu harus ditentukan MSE (Mean Square Error). MSE adalah nilai error kuadrat rata-rata [3].

$$PSNR = 10 \cdot \log_{10} \frac{(MAX)^2}{MSE} = 20 \cdot \log_{10} \frac{MAX}{\sqrt{MSE}} \tag{6}$$

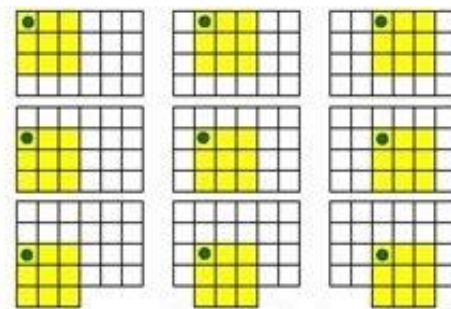
Dimana satuan PSRN dinyatakan dalam decibel, MAX merupakan nilai maksimal intensitas, yakni 255. Di mana MSE dirumuskan sebagai berikut:

$$MSE = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N [I(x, y) - I^1(x, y)]^2 \tag{7}$$

MN

$$y=1 \quad x=1$$

Di mana M=panjang citra (dalam pixel), N=lebar citra (dalam pixel) sedangkan $I(x, y)$ adalah nilai piksel dari citra cover dan $I^1(x, y)$ merupakan nilai piksel pada citra. Semakin rendah Nilai MSE maka akan semakin baik, dan semakin besar nilai PSNR maka semakin baik kualitas citra.



Gambar 3: Ilustrasi operasi konvolusi pada citra

- 5) PSNR(Peak Signal To Noise Ratio) dan MSE (Mean Square Error) PSNR (peak signal to noise ratio) merupakan nilai perbandingan antara harga maksimum dari citra hasil filtering dengan noise, yang dinyatakan dalam satuan desibel (dB). PSNR dapat digunakan untuk mengevaluasi nilai intensitas citra. Jika nilai

III. ANALISA DAN PEMBAHASAN

Pada pengujian system yang terdiri dari citra sumber dan citra target dikondisikan dalam keadaan grayscale. Pengujian terbagi menjadi video sumber menggunakan latar homogeny, video sumber berintensitas gelap dan video target gelap, video sumber berintensitas terang dan video target gelap, video sumber berintensitas gelap dan video target terang dan video sumber berintensitas terang dan video target terang dengan dimensi 320 x 240 piksel. Berikut beberapa video yang dikategorikan berlatar terang:

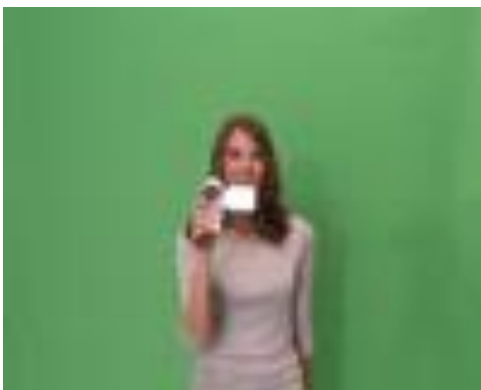


Gambar 4: Frame bagian dari video yang berlatar terang (1)

Video yang berlatar gelap:

Hasil dari pengujian data:

Hasil pengujian terhadap beberapa kategori video berdasarkan gelap terang latar video dihasilkan:



Gambar 5: Frame bagian dari video yang berlatar terang (2)



Gambar 6: Frame bagian dari video yang berlatar terang (3)



Gambar 7: Frame bagian dari video yang berlatar terang (4)



Gambar 8: Frame bagian dari video yang berlatar gelap (1)



Gambar 9: Frame bagian dari video yang berlatar gelap (2)



Gambar 10: Frame bagian dari video yang berlatar gelap (3)



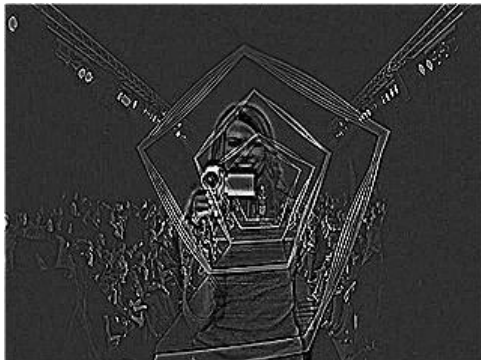
Gambar 14: Frame bagian dari video hasil (3)



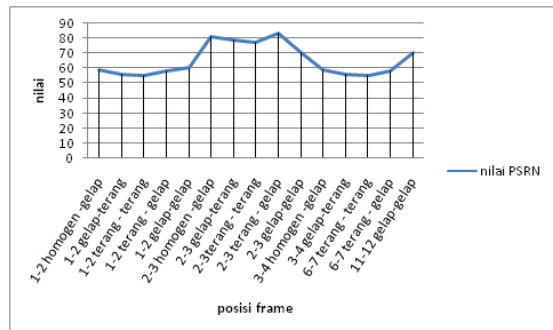
Gambar 11: Frame bagian dari video yang berlatar gelap (4)



Gambar 15: Frame bagian dari video hasil (4)



Gambar 12: Frame bagian dari video hasil (1)



Gambar 16: Grafis perbandingan antar frame sebagai berdasarkan parameter PSNR untuk mengetahui tingkat intensitas video



Gambar 13: Frame bagian dari video hasil (2)

IV. KESIMPULAN

Dari hasil pengujian didapat nilai PSNR terbaik pada video sumber berlatar gelap yang digabungkan dengan video target berlatar gelap dengan nilai PSNR mencapai 60, 0658 pada frame pertama dan kedua. Pada frame pertama video sumber dan target dioperasikan terhadap frame pertama matriks 3D poisson blending. Nilai intensitas berdasarkan PSNR pada frame 2 dan seterusnya cenderung naik kualitasnya hingga

lebih dari 10 desibel, sedangkan berdasarkan MSE nilainya turun hingga 0,1 sehingga disimpulkan intensitasnya cenderung lebih baik jika dibandingkan dengan frame pertama

V. DAFTAR PUSTAKA

- [1] P. P'erez, M. Gangnet, and A. Blake, "Poisson image editing," IEEE, 2003.
- [2] W. Handoko, E. Ardianto, and E. Safriliyanto, "Analisis dan implementasi image denoising dengan metode normal shrink sebagai wavelet thresholding analysis," Jurnal Teknologi Informasi DINAMIK, vol. Vol.16 no.1, pp. 56–63, 2011.
- [3] G. M. Male, Wirawan, and E. Setijadi, "Analisa kualitas citra pada steganografi untuk aplikasi e-government," Prosiding Seminar Nasional Manajemen Teknologi, vol. 15, 2012.

{Halaman sengaja dikosongkan}

{Jurnal **Melek IT**}

(4)

MEMBANGUN SERVER VOIP TRIXBOX DENGAN CLIENT ANDROID**Muhammad Faizal¹, Tjatur Widiartin²**Program Studi Teknik Informatika, Fakultas Teknik, Univesitas Wijaya Kusuma Surabaya
uwks09120039@gmail.com¹**ABSTRAK**

Server Voip Trixbox Adalah *VoIP Phone System* yang berbasiskan sistem *open source PBX (private Brance Exchange)* yang ketika sekali di-install kepada suatu PC dengan *interface* yang sesuai maka akan dapat di gunkan sebagai *full feature PBX* untuk pengguna rumahan, lembaga, dan lain sebagainya. Cara membangun *trixbox* dengan menginstall *trixbox* tersebut pada PC atau laptop sebagai *server* dan untuk mendapatkan nomor *caller id* tersebut kita harus mengkonfigurasikan *server* dan mendapatkan *caller id* pada *client*

Pada uji coba *voip* terbut menggukan android dimana dua android atau *client voip* yang akan digunakan uji coba menelfon pada dua android tersebut

Kata kunci : *Voip, Android*

PENDAHULUAN**Latar Belakang**

Saat ini jika seseorang saling berjangkauan atau berjauhan komunikasi maka komunikasi voice lah yang sering banyak digunakan pada banyak orang dengan menggunakan telepon, dengan menggunakan telepon kita bisa berkomunikasi dengan mudah walaupun dengan jarak yang berjauhan. Dan di dalam teknologi komunikasi, komunikasi suara merupakan satu hal yang akan menjadi bagian yang sangat penting. Karena saat ini komunikasi suara di anggap komunikasi yang paling praktis.

Berkomunikasi memang lebih mudah dan nyaman walaupun dengan jangkauan sangat jauh, akan tetapi dengan menggunakan telepon biasa kita bisa mendapatkan tagihan dengan biaya jangkauan jauh. bahkan ponsel dapat memadai untuk semua kebutuhan jarak jauh. Hal ini dapat membuat panggilan telepon mahal bagi kedua belah pihak .

Sebernarya untuk mengatasi masalah cost yang sangat tinggi telah ada solusinya yaitu bisa memanfaatkan teknologi VoIP. Dengan menggunakan VoIP ada beberapa keunggulan pada VoIP Dengan biaya yang murah penekanan dari voip dengan dua lokasi yang terhubung oleh internet sehingga biaya percakapan menjadi sangat rendah. Biaya lebih rendah untuk sambungan jarak jauh, memanfaatkan infrastruktur jaringan data yang sudah ada untuk suara. Berguna jika perusahaan sudah mempunyai jaringan. Jika memungkinkan jaringan yang ada bisa di bangun jaringan Voip dengan mudah. Tidak diperlukan tambahan biaya bulanan untuk penambahan komunikasi suara, penggunaan

bandwidth yang lebih kecil daripada telepon biasa. Dengan majunya teknologi penggunaan bandwidth untuk voice sekarang ini menjadi sangat kecil.

Penekanan utama VoIP, adalah kemudahan dalam berkomunikasi suara secara realtime melalui Internet. Jika kedua pengguna saling terhubung melalui Internet dan berada pada jarak yang saling berjauhan (antarkota dan antarnegara),Anda akan mendapatkan keuntungan dari segi biaya yang dikeluarkan. Dengan hanya membayar biaya Internet yang biasanya akan lebih murah daripada tarif telepon SLJJ (Sambungan Langsung Jarak Jauh) atau SLI (Sambungan Langsung Internasional), kedua pihak dapat saling berkomunikasi tanpa terbatas jarak.

TINJAUAN PUSTAKA**Pengertian VoIP**

Voice over Internet Protocol (VoIP) adalah teknologi yang memungkinkan membuat panggilan telepon dengan menggunakan koneksi internet broadband, bukan telepon biasa. Pada dasarnya, individu sekarang dapat menggunakan koneksi internet broadband mereka untuk menempatkan panggilan telepon daripada menggunakan saluran telepon tradisional mereka. Idealnya, kualitas suara adalah sama dengan jalur telepon standarTeknologi, walaupun kompleks, VoIP merupakan suatu teknologi komunikasi yang menawarkan kemudahan bertelepon melalui Internet. Kini berkomunikasi melalui telepon

tidak hanya bisa dilakukan dengan mengandalkan pesawat telepon konvensional maupun handphone saja, namun melalui Internet kita juga bisa melakukannya. Namun selain kelebihan yang ditawarkan oleh VoIP, terdapat kelemahan yang ada pada teknologi ini, misalnya kualitas suara yang dihasilkan tidak sejernih telepon biasa, dapat menyebabkan jaringan terhambat dal beberapa hal lain yang menjadi masalah untuk teknologi VoIP ini. Namun, perkembangan teknologi Voice over Internet Protocol sudah semakin pesat. Awalnya VoIP ini dianggap sepele tapi sekarang menjadi harapan pengguna sebagai alternatif telepon murah.

Sejarah Voip dan Penemu VOIP

Sejarah Perkembangan teknologi VoIP dimulai dari penemuan telepon pada tahun 1876 oleh [Alexander Graham Bell](#). Kemudian dikembangkan lagi teknologi [PSTN](#) (*Public Switched Telephone Network*) yang sudah berkembang sampai sekarang. Beberapa tahun kemudian mulai berkembang teknologi yang baru. Pembuatan *Personal Computer* (PC) secara massal, system komunikasi telepon selular dan terakhir system berdasarkan jaringan internet yang memberikan layanan e-mail, Chat dan lain-lain. Teknologi VoIP diperkenalkan setelah internet mulai berkembang sekitar tahun 1995. Pada mulanya kemampuan mengirimkan suara melalui internet hanya merupakan eksperimen dari beberapa orang atau perusahaan kecil. Ini dimulai dengan perusahaan seperti Vocaltech dan kemudian pada akhirnya diikuti oleh Microsoft dengan program Netmeeting-nya. Pada saat itu jaringan komputer internet masih sangat lambat. Di rumah-rumah (khususnya di Amerika) masih digunakan dial-up dengan kecepatan 36,6 Kbyte. Backbone Internet pun masih kecil. Aplikasi yang bersifat menghabiskan bandwidth, seperti misalnya suara atau video, masih sangat terbatas penggunaannya di pusat penelitian yang memiliki bandwidth besar.

Android

Beberapa tahun belakangan ini istilah Android sering sekali kita dengar, baca maupun kita lihat. Pada umumnya istilah Android sering dikaitkan dengan Ponsel, smartphone

dan tablet. Penemu Android adalah Andy Rubin yang lahir pada tanggal 22 Juni 1946 di New Bedford, Amerika Serikat. Andy Rubin bersama-sama dengan Rich Miner, Nick Sears, dan Chris White mendirikan Android.inc dan pada Juli 2005 dibeli oleh Google.

Android adalah sistem operasi berbasis Linux yang dipergunakan sebagai pengelola sumber daya perangkat keras, baik untuk ponsel, smartphone dan juga PC tablet. Secara umum Android adalah platform yang terbuka (Open Source) bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh berbagai piranti bergerak. Telepon pertama yang memakai sistem operasi Android adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2009 diperkirakan di dunia ini paling sedikit terdapat 18 jenis telepon seluler yang menggunakan Android. Semenjak kehadirannya pada 9 Maret 2009, Android telah hadir dengan versi 1.1, yaitu sistem operasi yang sudah dilengkapi dengan pembaruan estetis pada aplikasinya, seperti jam alm, voice search, pengiriman pesan dengan Gmail, dan pemberitahuan email Hingga tahun 2012, Android telah berkembang dengan pesat. Dalam kurun 3 tahun Android telah diproduksi dalam versi, dan versi terakhir yang diproduksi disebut sebagai Android versi 4.1 atau Android Jelly Bean.

PC

Personal Computer atau PC merupakan jenis komputer yang paling banyak digunakan saat ini. PC yang dirancang untuk single user (pemakai tunggal) sebenarnya diperkenalkan oleh IBM pertama kali pada tahun 1981. Bila generasi sebelumnya, bentuk komputer sangatlah besar sehingga tidak praktis, maka sejak ditemukannya chip pada tahun 1971, ukuran komputer menjadi semakin kecil. Selain lebih praktis, harga dari PC juga jauh lebih murah bila dibandingkan dengan computer sebelumnya. Computer adalah alat yang dipakai untuk mengolah [data](#) menurut perintah yang telah dirumuskan. Kata komputer semula dipergunakan untuk menggambarkan orang yang perkerjaannya melakukan [perhitungan aritmatika](#), dengan atau tanpa alat bantu, tetapi arti kata ini kemudian dipindahkan kepada [mesin](#) itu sendiri. Asal mulanya, pengolahan informasi hampir eksklusif berhubungan dengan masalah [aritmatika](#), tetapi komputer modern dipakai untuk banyak tugas yang tidak berhubungan dengan [matematika](#). Secara

luas, Komputer dapat didefinisikan sebagai suatu peralatan elektronik yang terdiri dari beberapa komponen, yang dapat bekerja sama antara komponen satu dengan yang lain untuk menghasilkan suatu informasi berdasarkan program dan data yang ada. Adapun komponen komputer adalah meliputi : Layar Monitor, CPU, Keyboard, Mouse dan Printer (sbg pelengkap). Tanpa printer komputer tetap dapat melakukan tugasnya sebagai pengolah data, namun sebatas terlihat dilayar monitor belum dalam bentuk print out (kertas). Dalam definisi seperti itu terdapat alat seperti [slide rule](#), jenis [kalkulator mekanik](#) mulai dari [abakus](#) dan seterusnya, sampai semua komputer elektronik yang [kontemporer](#). Istilah lebih baik yang cocok untuk arti luas seperti "komputer" adalah "yang [memproses informasi](#)" atau "[sistem pengolah informasi](#)." Saat ini, komputer sudah semakin canggih. Tetapi, sebelumnya komputer tidak sekecil, secanggih, sekeren dan seringan sekarang. Dalam sejarah komputer, ada 5 generasi dalam sejarah komputer.

Foto: www.fast-computer.webs.com



Gambar 2.1 PC

Jaringan Komputer

Jaringan komputer (jaringan) adalah sebuah [sistem](#) yang terdiri atas [komputer](#)-komputer yang didesain untuk dapat berbagi sumber daya (printer, CPU), berkomunikasi (surel, pesan instan), dan dapat mengakses informasi (peramban web). Tujuan dari jaringan komputer adalah agar dapat mencapai tujuannya, setiap bagian dari jaringan komputer dapat meminta dan memberikan layanan (service). Pihak yang meminta/menerima layanan disebut klien ([client](#)) dan yang memberikan/mengirim layanan disebut peladen ([server](#)). Desain ini disebut dengan sistem [client-server](#), dan digunakan pada hampir seluruh [aplikasi](#) jaringan komputer.

Jaringan Wilayah Lokal (LAN)

adalah jaringan [komputer](#) yang jaringannya hanya mencakup wilayah kecil: seperti jaringan kampus, gedung, kantor dalam rumah, sekolah atau yang lebih kecil. Saat ini, kebanyakan LAN berbasis pada teknologi IEEE 802.3 Ethernet menggunakan perangkat switch, yang mempunyai kecepatan transfer data 10, 100, atau 1000 Mbit/s. Selain teknologi Ethernet, saat ini teknologi 802.11b (atau biasa disebut Wi-fi) juga sering digunakan untuk membentuk LAN. Tempat-tempat yang menyediakan koneksi LAN dengan teknologi Wi-fi biasa disebut hotspot. Pada sebuah LAN, setiap node atau komputer mempunyai daya komputasi sendiri, berbeda dengan konsep dump terminal. Setiap komputer juga dapat mengakses sumber daya yang ada di LAN sesuai dengan hak akses yang telah diatur. Sumber daya tersebut dapat berupa data atau perangkat seperti printer. Pada LAN, seorang pengguna juga dapat berkomunikasi dengan pengguna yang lain dengan menggunakan aplikasi yang sesuai.

Berbeda dengan Jaringan Area Luas atau Wide Area Network (WAN), maka LAN mempunyai karakteristik sebagai berikut :

1. Mempunyai pesat data yang lebih tinggi
2. Meliputi wilayah geografi yang lebih sempit
3. Tidak membutuhkan jalur telekomunikasi yang disewa dari operator telekomunikasi

LAN tersusun dari beberapa elemen dasar yang meliputi komponen hardware dan software, yaitu:

1. Komponen Fisik Personal Computer (PC), Network Interface Card (NIC), Kabel, Topologi jaringan.
2. Komponen Software Sistem Operasi Jaringan, Network Adapter Driver, Protokol Jaringan. Jaringan ini disebut sebagai jaringan area, yaitu jaringan yang terbatas untuk area kecil, seperti pada lingkungan perkantoran di sebuah gedung, sekolah, atau kampus. Dalam jaringan LAN, terdapat satu komputer yang biasa disebut server, yang fungsinya adalah untuk memberikan layanan perangkat lunak (software), mengatur aktivitas jaringan dan menyimpan file. Selain server ada pula komputer lain yang terhubung dalam jaringan (network) yang disebut dengan workstation (client). Pada umumnya teknologi jaringan LAN menggunakan media kabel untuk menghubungkan komputer-komputer yang digunakan.
3. LAN dapat dibedakan berdasarkan tiga karakteristik, yaitu ukuran, teknologi transmisi, dan topologinya. Teknologi transmisi yang bisa digunakan adalah transmisi kabel tunggal. Pada LAN biasa, kecepatan transmisi sekitar 10 – 100 Mbps

(Megabit/second), dan faktor kesalahan kecil. Topologi yang digunakan biasanya topologi Bus, Star dan Ring.

Jaringan Nirkabel

adalah bidang disiplin yang berkaitan dengan komunikasi antar sistem komputer tanpa menggunakan kabel. Jaringan nirkabel ini sering dipakai untuk jaringan komputer baik pada jarak yang dekat (beberapa meter, memakai alat/pemancar bluetooth) maupun pada jarak jauh (lewat satelit). Bidang ini erat hubungannya dengan bidang telekomunikasi, teknologi informasi, dan teknik komputer. Jenis jaringan yang populer dalam kategori jaringan nirkabel ini meliputi: Jaringan kawasan lokal nirkabel (wireless LAN/WLAN), dan Wi-Fi. Jaringan nirkabel biasanya menghubungkan satu sistem komputer dengan sistem yang lain dengan menggunakan beberapa macam media transmisi tanpa kabel, seperti: gelombang radio, gelombang mikro, maupun cahaya infra merah.

Sejarah Jaringan Nirkabel

Pada tahun 1970 Norman Abramson, seorang profesor di University of Hawaii, mengembangkan komputer pertama di dunia jaringan komunikasi, ALOHAnet, menggunakan biaya rendah seperti ham-radio. Dengan bi-directional topologi bintang, sistem komputer yang terhubung tujuh ditempatkan lebih dari empat pulau untuk berkomunikasi dengan komputer pusat di Pulau Oahu tanpa menggunakan saluran telepon. "Pada tahun 1979, FR Gfeller dan U. Bapst menerbitkan makalah di Proceedings IEEE pelaporan percobaan jaringan area lokal nirkabel menggunakan komunikasi infra merah disebarkan. Tak lama kemudian, pada tahun 1980, P. Ferrert melaporkan percobaan penerapan kode satu radio spread spectrum untuk komunikasi di terminal nirkabel IEEE Konferensi Telekomunikasi Nasional. Pada tahun 1984, perbandingan antara infra merah dan CDMA spread spectrum untuk komunikasi jaringan informasi kantor nirkabel diterbitkan oleh IEEE Kaveh Pahlavan di Jaringan Komputer Simposium yang muncul kemudian dalam IEEE Communication Society Magazine. Pada bulan Mei 1985, upaya Marcus memimpin FCC untuk mengumumkan ISM band eksperimental untuk aplikasi komersial teknologi spread spectrum. Belakangan, M. Kavehrad melaporkan percobaan sistem PBX nirkabel kode menggunakan Division Multiple Access. Upaya-upaya ini mendorong kegiatan industri yang signifikan dalam pengembangan dari generasi baru dari jaringan area lokal nirkabel dan diperbarui beberapa lama diskusi

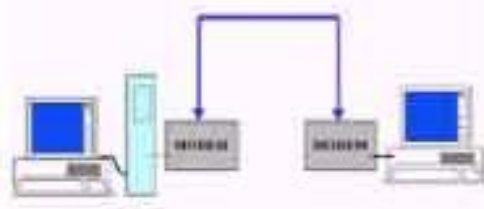
di radio portabel dan mobile industri. Generasi pertama dari modem data nirkabel dikembangkan pada awal 1980-an oleh operator radio amatir, yang sering disebut sebagai radio paket ini. Mereka menambahkan komunikasi data pita suara modem, dengan kecepatan data di bawah 9.600-bit / s, untuk yang sudah ada sistem radio jarak pendek, biasanya dalam dua meter band amatir. Generasi kedua modem nirkabel dikembangkan FCC segera setelah pengumuman di band eksperimental untuk non-militer penggunaan spektrum penyebaran teknologi. Modem ini memiliki kecepatan data yang diberikan atas perintah ratusan kbit / s. Generasi ketiga modem nirkabel ditujukan untuk kompatibilitas dengan LAN yang ada dengan data tingkat atas perintah Mbit / s. Beberapa perusahaan yang mengembangkan produk-produk generasi ketiga dengan kecepatan data diatas 1 Mbit / s dan beberapa produk sudah diumumkan oleh waktu pertama IEEE Workshop on Wireless LAN.

Jaringan Topologi

adalah, hal yang menjelaskan hubungan geometris antara unsur-unsur dasar penyusun jaringan yaitu [node](#), [link](#), dan [station](#).

Topologi Point to Point

Jaringan kerja titik ke titik merupakan jaringan kerja yang paling sederhana tetapi dapat digunakan secara luas. Begitu sederhananya jaringan ini, sehingga seringkali tidak dianggap sebagai suatu jaringan tetapi hanya merupakan komunikasi biasa. Dalam hal ini, kedua simpul mempunyai kedudukan yang setingkat, sehingga simpul manapun dapat memulai dan mengendalikan hubungan dalam jaringan tersebut. Data dikirim dari satu simpul langsung kesimpul lainnya sebagai penerima, misalnya antara terminal dengan CPU.



Gambar 2.2 Topologi Point to Point Topologi Plex Network (Jaringan Kombinasi)

Merupakan jaringan yang benar-benar interactive, dimana setiap simpul mempunyai kemampuan untuk meng-access secara langsung tidak hanya terhadap komputer, tetapi juga dengan peralatan ataupun simpul yang lain. Secara umum, jaringan ini mempunyai bentuk mirip dengan jaringan bintang. Organisasi data yang ada

menggunakan de-sentralisasi, sedang untuk melakukan perawatan, digunakan fasilitas sentralisasi



Gambar 2.3 Topologi Plex Network (Jaringan Kombinasi)

Trixbox

Trixbox (Asterisk@home) adalah VoIP Phone System yang berbasis sistem open source PBX (private Branch Exchange) yang ketika sekali di-install kepada suatu PC dengan interface yang sesuai maka akan dapat di gunakan sebagai full feature PBX untuk pengguna rumahan, lembaga, dan lain sebagainya. Trixbox sangat terkenal karena mengkombinasikan paket-paket Open Source Telepon terbaik yang disertakan di dalam sistem operasi tersebut. Dahulu seorang user dituntut untuk menjadi seorang programmer jika ingin mengimplementasikan VoIP karena User Interface yang tidak Friendly. oleh karena itu untuk mengatasi masalah seperti itu diluncurkannya Trixbox dengan penggunaan yang sangat mudah karena menu utama yang berbasis web browsure untuk menkonfigurasi dan mengatur sistem, serta pasket-paket untuk VoIP Server dijadikan dalam satu bundle dengan operating system CentOS sehingga menjadi sistem Trixbox CE. Trixbox dibuat oleh Andrew Gillis pada bulan november 2004 dengan tujuan untuk membuat para pengguna komputer biasa dapat menggunakan secara maksimal Asterisk PBX system tanpa dibutuhkannya pengajar atau pengetahuan lebih mengenai VoIP. Sebelumnya Trixbox menggunakan nama Asterisk@home, namun dikarenakan Asterisk merupakan nama dagang dari perusahaan Digium.Ltd dan @home tidak sesuai dengan fungsionalitas dari Trixbox yang dapat melayani lebih dari sekedar pengguna rumahan atau bisnis sekala kecil dan menengah.

ANALISA DAN PERANCANGAN

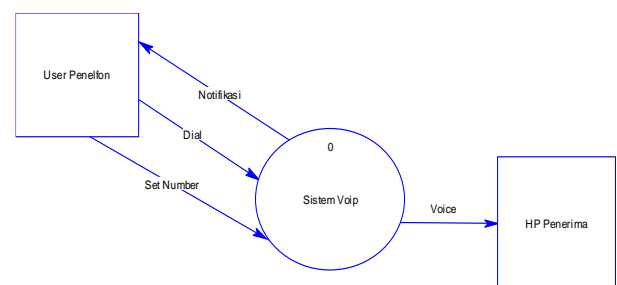
Analisa Kebutuhan Sistem

Sistem teknologi voice menggunakan teknologi voip adalah sistem yang bisa digunakan untuk

berkomunikasi suara antara dua personal. Teknologi yang di bangun ini memakai 2 android client, dimana masing komputer harus di seting terlebih dahulu dengan software voip. Selajutnya android client yang akan di gunakan untuk berkomunikasi harus terhubung dengan jaringan internet.

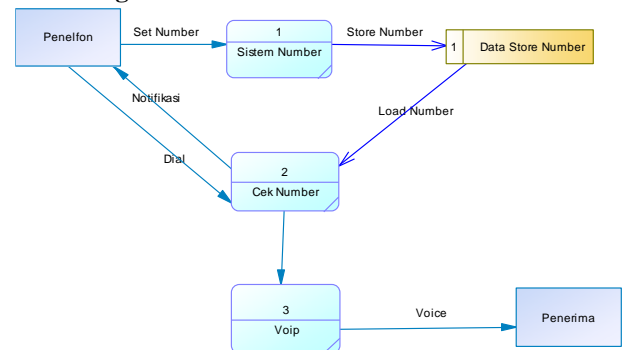
Cara Berkomunikasi

Pada gambar 3.2 menjelaskan analisa sistem dan laju kualitas voip menjelaskan user voip menseset nomor dan di proses ke sistem voip, sistem voip akan memberikan notifikasi nomor telah terdaftar setelah itu user penelpon akan mendial dan akan di proses pada sistem voip dan sistem voip akan memberi output voice ke HP atau user penerima



Gambar 3.1 Diagram Konteks analisa sistem dan laju kualitas sistem VoIP

3.1.2 Diagram Level 0 VoIP



Gambar 3.2 Diagram Level 0 VoIP

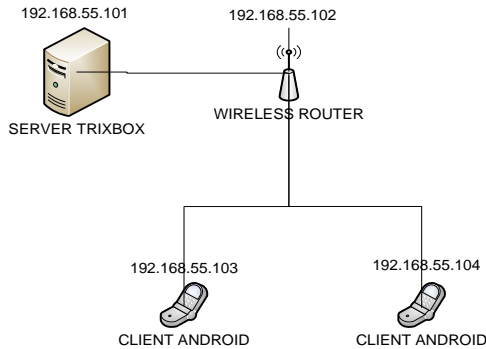
Pada gambar 3.2 menjelaskan bisnis proses dimana penelpon akan menseset nomor ke sistem nomor dan akan di store number ke database dan store number yang didatabase meload ke sistem cek number dan sistem cek number akan memberikan notifikasi pada user penelpon jika nomor telah terdaftar setelah itu user penelpon akan mendial ke sistem cek number, dan sistem cek number akan mentransfer data ke sistem voip dan sistem voip mengeluarkan output data voice ke user penerima

Desain Sistem

Berdasarkan gambaran sistem yang ada di uraikan dibagian analisa sistem selajutnya di ulas desain untuk sistem VoIP

Desain Jaringan

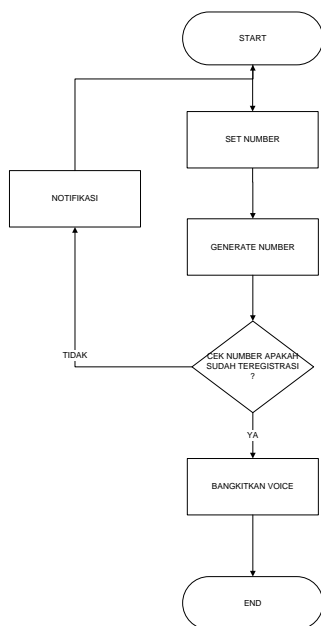
Pada Gambar 3.3 dijelaskan dari desain jaringan membangun server voip trixbox dengan client android bahwa server trixbox dengan ip 192.168.55.101 terhubung pada wireless router yang akan terhubung pada client handphone android



Gambar 3.2 Desain Jaringan

Algoritma Proses Konfigurasi Server

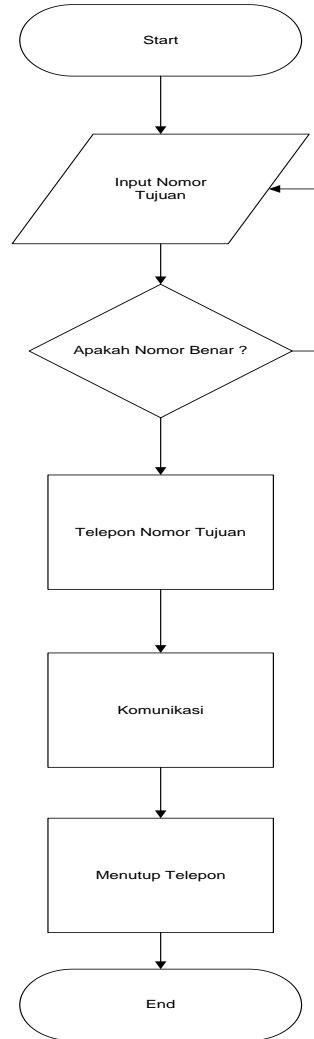
Pada gambar 3.4 yaitu menjelaskan tentang jalannya proses algoritma proses dimulai start lalu masuk ke inputan menset number, setelah inputan menset number masuk ke proses generate number dan setelah memproses proses generate number masuk ke decision apakah pengecekan number sudah terregistrasi atau belum jika tidak akan masuk proses notifikasi dan masuk ke proses menset number jika terdaftar atau ya akan masuk ke proses voice dan selesai



Gambar 3.3 Algoritma Proses Konfigurasi Server

Algoritma Proses Penelfonan

Pada gambar 3.5 yaitu menjelaskan tentang jalannya proses algoritma proses dimulai start dengan menginputkan nomor tujuan setelah menginputkan nomortujuan akan memasuki proses validasi nomor tujuan dan akan memasuki decision apakah nomor yang dimasukan benar jika tidak akan mengulang pada inputan nomor tujuan jika benar memasuki proses telepon nomor tujuan dan memasuki proses komunikasi berjalan dan proses penutupan telepon



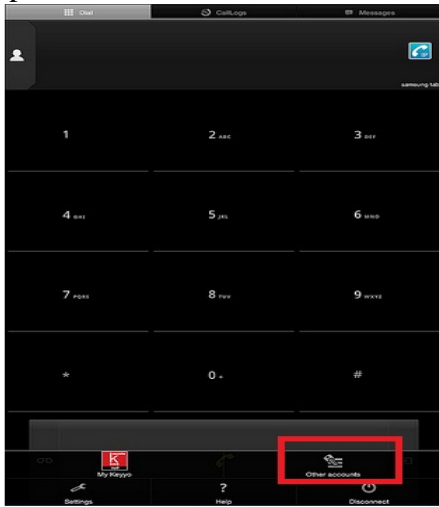
Gambar 3.4 Algoritma Proses Konfigurasi Client

IMPLEMENTASI PERANGKAT LUNAK

Uji Coba Sistem

Dalam melakukan pengujian sistem, yang dilakukan dalam membangun server voip trixbox dengan client android

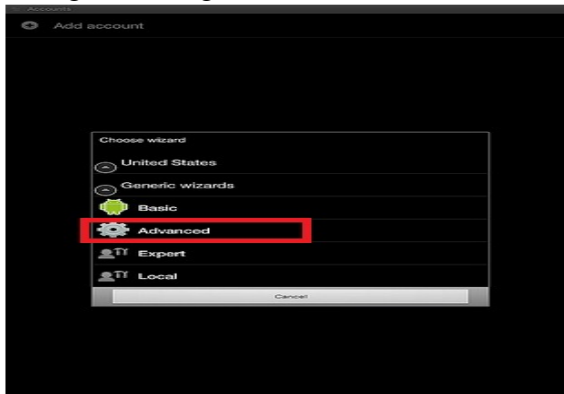
seperti dijelaskan pada sub bab 1,2,3, dan 4 telah menghasilkan beberapa skenario melewati :



Gambar 5.1 Aplikasi Keyyo Voip

Keterangan :

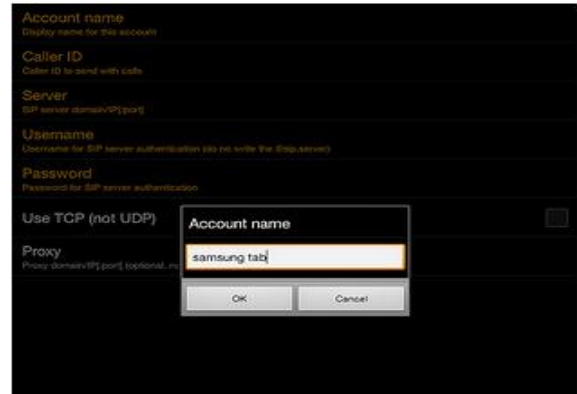
Pada Gambar 5.1 ini menjelaskan aplikasi keyyo voip terbuka dengan penyettingan konfigurasi dengan klik *other account*



Gambar 5.2 Tampilan Aplikasi Keyyo Other Account Setting

Keterangan :

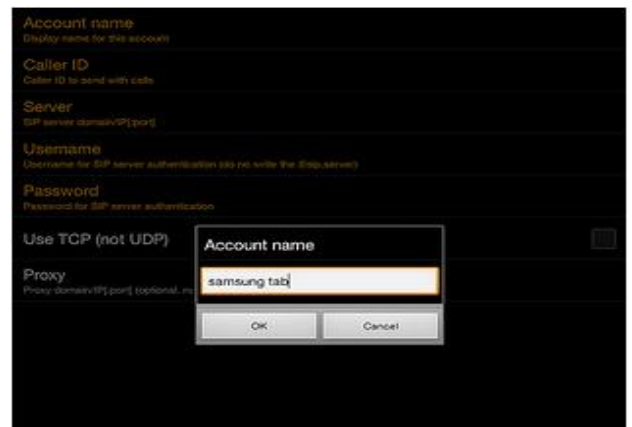
Pada Gambar 5.2 Setelah tampilan *Other Account Setting* muncul kita lanjutkan dengan *Add Account* kemudian klik advance



Gambar 5.3 Halaman Setting add Account Name

Keterangan :

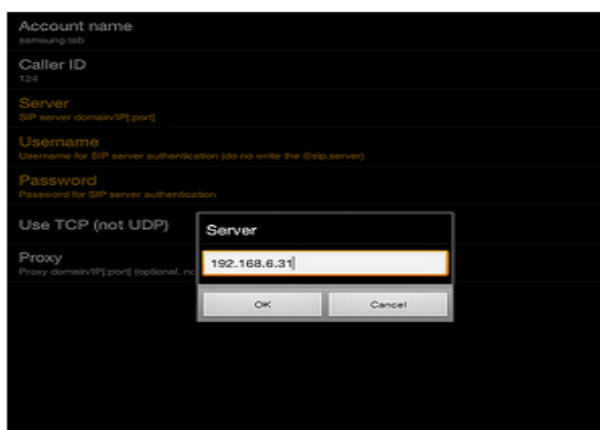
Pada Gambar 5.3 kami menginputkan account name yang sudah didaftarkan pada server, dan Account name pengisiannya harus sama pada server, jika Account name yang di isi berbeda maka Account name yang akan di daftarkan tidak akan terdaftar



Gambar 5.4 Tampilan entri Caller ID

Keterangan :

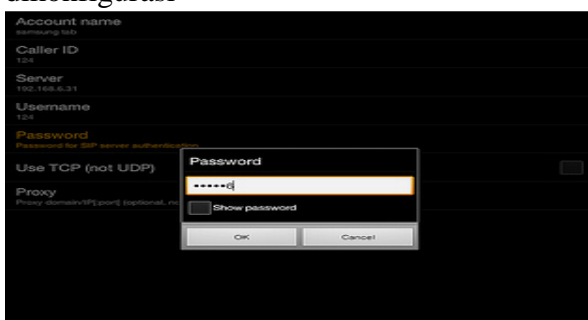
Pada Gambar 5.4 pada form Caller ID di persilakan untuk mengisi nomor 124 dimana konfigurasi tersebut untuk mempermudah pemanggilan pada client lain dan konfigurasi ini harus menginputkan sama pada server yang sudah di daftarkan pada konfigurasi server



Gambar 5.5 Tampilan Entri Server

Keterangan :

Pada Gambar 5.5 entri server isikan alamat server yang sudah di daftarkan pada server dan sudah di konfigurasi setelah kami menginputkan server selanjutnya menginputkan username yang sudah didaftarkan pada server dan sudah dikonfigurasi



Gambar 5.6 Tampilan entri Password

Keterangan :

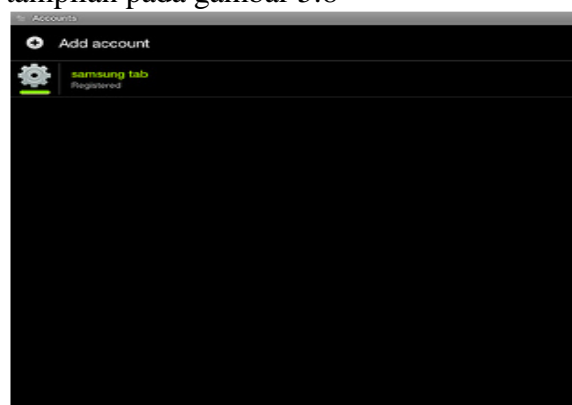
Pada Gambar 5.6 di tampilan entri password kita mengisikan nomor “ 123456 “ yang sudah terdaftar pada server jika password tidak benar maka untuk mamnggilan voip kepada client lain tidak akan bisa



Gambar 5.7 Penyelesaian Setting Keyyo

Keterangan :

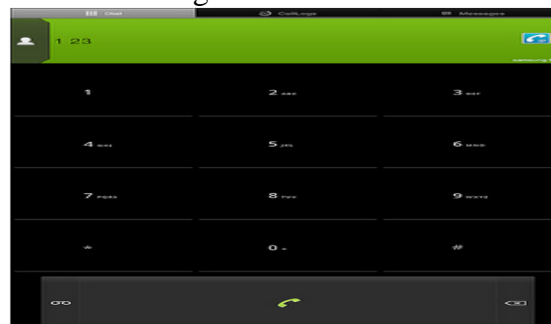
Pada Gambar 5.7 setelah pengisian keyyo selesai maka klik ok dan akan muncul tampilan pada gambar 5.8



Gambar 5.8 Status Register Pada Penyetingan Keyyo

Keterangan :

Pada Gambar 5.8 ketika semua pengisian di atas tidak ada masalah, maka akan muncul kalimat “ Registered “ seperti pada gambar 5.8 yang artinya gadget kita telah terkoneksi dengan baik



Gambar 5.9 Interface Client Keyyo Voip

Keterangan :

Pada gambar 5.9 Tampilan untuk berkomunikasi voice kepada client



Gambar 5.10 IP Android Smartfren A

Keterangan :

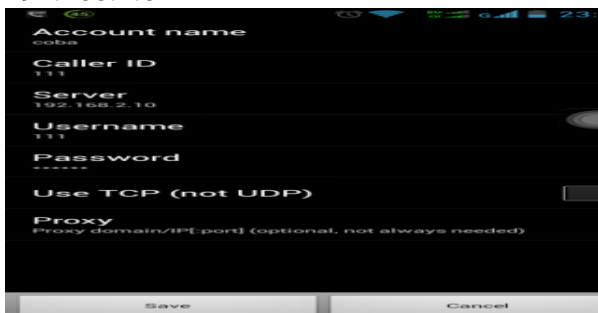
Pada gambar 5.10 Voip menggunakan android smartfren dengan android smartfren. Koneksikan android smartfren keduanya pada acces point TP-Link. android smartfren terhubung pada acces point dengan IP 192.168.2.4



Gambar 5.11 IP Android Smartfren B

Keterangan :

Pada gambar 5.11 IP android smartfren terhubung pada acces point dengan IP 192.168.2.6



Gambar 5.12 Konfigurasi Smartfren A

Keterangan :

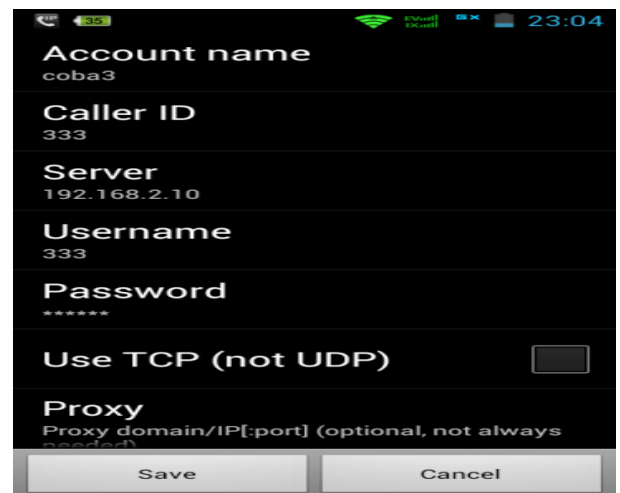
Pada gambar 5.12 android smartfren A mendapatkan account nama coba dengan caller id 111 dengan memasukan nomor IP server 192.168.2.10 dan menuliskan username 111 dan menuliskan password 123456 dengan menyamakan konfigurasi pada server



Gambar 5.13 Konfigurasi Samsung

Keterangan :

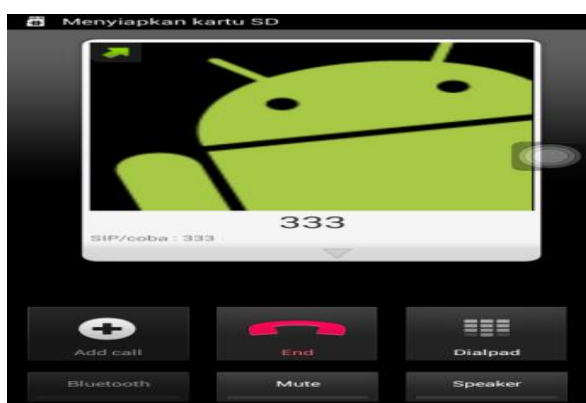
Pada gambar 5.13 android samsung mendapatkan account nama coba dengan caller id 333 dengan memasukan nomor IP server 192.168.2.10 dan menuliskan username 333 dan menuliskan password 123456 dengan menyamakan konfigurasi pada server



Gambar 5.14 Konfigurasi Smartfren B

Keterangan :

Pada gambar 5.14 android smartfren B mendapatkan account nama coba dengan caller id 333 dengan memasukan nomor IP server 192.168.2.10 dan menuliskan username 333 dan menuliskan password 123456 dengan menyamakan konfigurasi pada server. Setelah terkoneksi android smartfren A dan B terhadap acces point, maka akan dilakukan komunikasi voice dari android smartfren A terhadap smartfren B.



Gambar 5.15 Pemanggilan Pada Smartfren B

Keterangan :

Pada gambar 5.15 melakukan panggilan komunikasi voice kepada android smartfren B dengan nomor sip 333



Gambar 5.16 penerima pada smartfren A

Keterangan :

Pada gambar 5.16 melakukan penerimaan panggilan komunikasi voice kepada android smartfren A dengan nomor sip 111

Perangkat	Sinyal	Jarak AP	Delay
Android Smartfren A Terhadap android Smart freen B	Kuat	1 Meter	2 Second
	Sedang	4 Meter	4 Second
	Lemah	7 Meter	8 Second

Tabel 5.1 Hasil Analisa pada Android smartfren A ke smartfren B

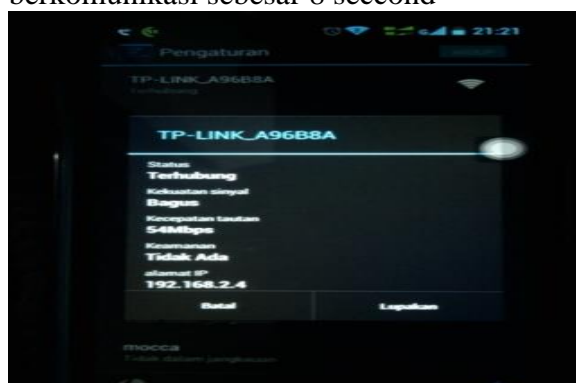
Keterangan :

Setelah bisa melakukan komunikasi voice maka tahap selanjutnya dilakukan komunikasi voice berdasarkan jarak untuk mengetahui kekuatan sinyal dan delay pada saat melakukan komunikasi voice android smartfren A android smartfren B

Pada Tabel 5.1 Smartfren A dengan sinyal kuat pada jarak 1 meter delay pada saat berkomunikasi sebesar 2 second

Pada Tabel 5.1 Smartfren A dengan sinyal kuat pada jarak 4 meter delay pada saat berkomunikasi sebesar 4 second

Pada Tabel 5.1 Smartfren A dengan sinyal kuat pada jarak 7 meter delay pada saat berkomunikasi sebesar 8 second

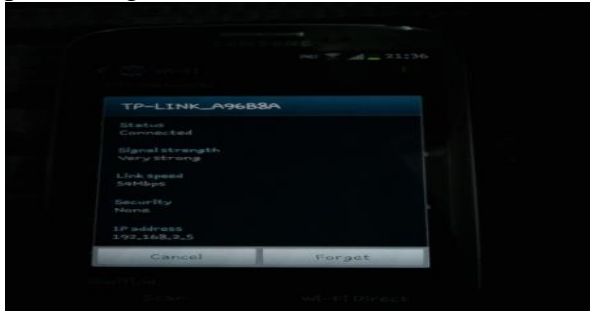


Gambar 5.17 IP Android Smartfren

Keterangan :

Voip menggunakan android smartfren dengan android Samsung. Koneksikan android smartfren dan Samsung pada acces point TP-Link. Pada gambar 5.17 IP

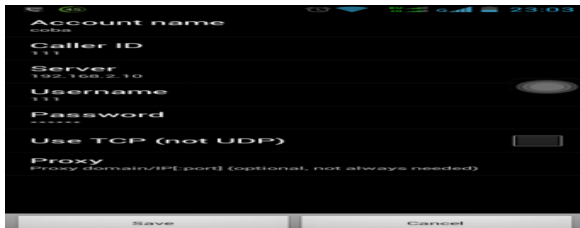
android smartfren terhubung pada acces point dengan IP 192.168.2.4



Gambar 5.18 IP Android Samsung

Keterangan :

Pada gambar 5 IP android Samsung terhubung pada acces point dengan IP 192.168.2.5



Gambar 5.19 Konfigurasi Smartfren

Keterangan :

Pada gambar 5.19 android smartfren A mendapatkan account nama coba dengan caller id 111 dengan memasukan nomor IP server 192.168.2.10 dan menuliskan username 111 dan menuliskan password 123456 dengan menyamakan konfigurasi pada server.



Gambar 5.20 Konfigurasi Samsung

Keterangan :

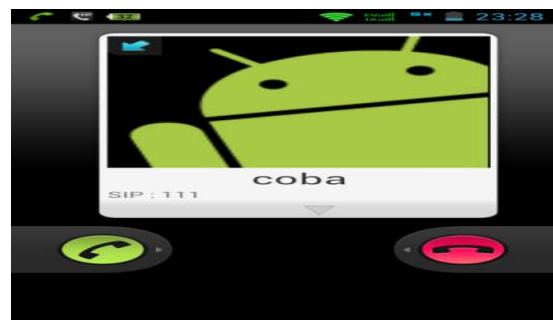
Pada gambar 5.20 android samsung mendapatkan account nama coba3 dengan caller id 222 dengan memasukan nomor IP server 192.168.2.10 dan menuliskan username 222 dan menuliskan password 123456 dengan menyamakan konfigurasi pada server. Setelah terkoneksi android smartfren dan Samsung terhadap acces point, maka akan dilakukan komunikasi voice dari android smartfren terhadap Samsung .



Gambar 5.21 Pemanggilan Pada Samsung

Keterangan :

Pada gambar 5.21 melakukan panggilan komunikasi voice kepada android Samsung dengan nomor sip 333



Gambar 5.22 Penerimaan Panggilan Pada Smartfren

Keterangan :

Pada gambar 5.22 melakukan penerimaan panggilan komunikasi voice android smartfren A dengan nomor sip 111

Perangkat	Sinyal	Jarak AP	Delay
Android Smartfren Terhadap android Smart Samsung	Kuat	1 Meter	2 Second
	Sedang	4 Meter	3 Second
	Lemah	7 Meter	6 Second

Tabel 5.1 Hasil Analisa pada Android smartfren ke Samsung

Keterangan :

Setelah bisa melakukan komunikasi voice maka tahap selanjutnya dilakukan komunikasi voice berdasarkan jarak untuk mengetahui kekuatan sinyal dan delay pada saat melakukan komunikasi voice android smartfren A android smartfren B

Pada Tabel 5.1 Smartfren dengan sinyal kuat pada jarak 1 meter delay pada saat berkomunikasi sebesar 2 second

Pada Tabel 5.1 Smartfren dengan sinyal kuat pada jarak 4 meter delay pada saat berkomunikasi sebesar 3 second.

PENUTUP

Kesimpulan

Dari hasil implementasi yang telah dilakukan serta uji coba terhadap fungsionalitas serta performa dari aplikasi voipServer (Server) dan voip (Client) yang telah dibangun dapat disimpulkan bahwa :

1. Dengan menggunakan Trixbox Sebagai Server dan android sebagai client dapat melakukan komunikasi voice pada jaringan Local
2. Delay pada saat berkomunikasi ditentukan dengan kekuatan sinyal android pada access point
3. Delay pada saat berkomunikasi sesama android smartfren hanya sebesar 6s, smartfren dan Samsung sebesar 8s lebih besar dibandingkan sesama produk
4. Aplikasi client pada Android telah berhasil dibangun sehingga memudahkan adanya komunikasi antar client melalui VoIP.

Saran

Berikut merupakan beberapa saran untuk pengembangan sistem di masa yang akan datang, berdasar pada hasil perancangan, implementasi, dan uji coba yang telah dilakukan.

1. Client bisa dikembangkan lebih lanjut untuk menampilkan user yang sedang online.
2. Ke depannya aplikasi Client dikembangkan untuk sistem layanan conference call, dengan sistem ini komunikasi dapat berlangsung dengan baik dalam melayani komunikasi dengan banyak client.
3. Adanya delay pada aplikasi ini menjadi catatan tersendiri, sehingga pada pengembangan selanjutnya perlu dilakukan analisa pada teknik kompresi suara agar suara yang dikirim lebih ringan dan lebih cepat, sehingga delay dapat diperkecil.

DAFTAR PUSTAKA

Anonymous,(2013),Pengertian dan Definisi Voip, Diakses pada 12 Mei 2013,<http://id.wikipedia.org/wiki/Voip>

Anonymous,(2013), Cara Kerja Voip, Diakses pada 2 Mei 2013<http://nanaafriana.blogspot.com/2011/04/pengertianvoip.html>

Anonymous,(2013), Sejarah Wireless, Diakses Pada 10 Mei 2013<http://wikarahayu34xltkj2.blogspot.com/2012/06/sejarahrouter.html>

Anonymous,(2013), Voip dan Kegunaannya, Diakses pada 28Mei2013<http://volkshymne.blogspot.com/2010/02/pengertianvoip.html>

Onno W. Purbo, 2007. Voip Untuk Rakyat Cikal Bakal Telkom Rakyat, Jakarta : CenterOf ICT studies

(8)

**PERGERAKAN KERUMUNAN MENGGUNAKAN RVO (*RECIPROCAL VELOCITY OBSTACLE*)
WITH EXTENDED ORCA (*OPTIMAL RECIPROCAL COLLISION AVOIDANCE*) UNTUK
NPC (*NON PLAYER CHARACTER*).**

Fariz Djunaidi,ST¹, Mochamad Hariadi, ST,Msc,Phd.², Dr.Supeno mardi S.N,ST.MT²

¹Student at Teknik Komputer & Telematika, Jurusan Teknik Elektro ITS Surabaya

²Lecture at Teknik Komputer & Telematika, Jurusan Teknik Elektro ITS Surabaya

ABSTRAK

Istilah “kerumunan” menurut Le Bon, seorang ahli sosiologi dalam bukunya yang berjudul “*The crowd, a study of popular mind*” adalah sejumlah individu yang berkumpul bersama dengan haluan yang sama, Individu agent yang bersama- sama dengan jarak yang dekat akan memperlihatkan tingkah lakunya yang bersifat konvergensi, yaitu keseragaman maksud, tindakan dan tujuan antara individu yang satu dan yang lain.

Pada penelitian ini, dibuat sebuah pergerakan kerumunan yang memiliki konvergensi antar agen yang satu dengan agen yang lain dengan memanfaatkan model yang berbeda-beda. Pergerakan kerumunan sangat bermanfaat bila digunakan sebagai kerumunan NPC (*Non Player Character*) yang cerdas pada suatu simulasi atau permainan. Kecerdasan yang diimplementasikan pada setiap agen ini menggunakan algoritma RVO (*Reciprocal Velocity Obstacle*), sebuah algoritma yang menitikberatkan pada kemampuan agen untuk mengunci tujuan pada goal yang diinginkan dan kecepatan agen dalam merespon daya hindar sehingga mencegah terjadinya tabrakan (*Collision Avoidance*), baik antara agen dengan agen yang lain dalam satu kerumunan maupun dengan kerumunan yang lain.

Pengujian dilakukan dengan membuat beberapa skenario, dimana model dalam satu kerumunan bertemu dengan kerumunan lain di jalan terbuka, jalan tertutup, dan perempatan jalan. Pada setiap pengujian, jumlah kerumunan dibuat bervariasi mulai dari kerumunan yang terdiri dari 25 agent sampai 1200 agent. Hasil dari pengujian menunjukkan bahwa kerumunan A dan B mampu mengurai beberapa saat setelah agen A berkerumun dengan agen B, sampai kerumunan tersebut terurai, dan setiap kerumunan sampai pada tujuan masing-masing. Selain skenario diatas, pengujian juga dilakukan terhadap kecepatan antar kerumunan terhadap kerumunan yang lain. Hasil menunjukkan bahwa pergerakan bisa diimplementasikan untuk memodelkan kerumunan yang bergerak dari satu titik ke titik yang lain, dan kemampuan daya hindar pada kerumunan tersebut.

Kata Kunci: RVO (*Reciprocal Collision Obstacle*), *Collision Avoidance*, *Agent*, NPC.

I. PENDAHULUAN

Pada dekade belakangan ini seiring dengan bertumbuhnya teknologi komputer dan seiring pula dengan berkembangnya metode-metode simulasi maka kerumunan menjadi topik hangat yang terus diangkat untuk dikembangkan, diperbaiki, dan diimplementasikan untuk kepentingan simulasi evakuasi korban, strategi taktis untuk exit plan, dan pembuatan game-game yang didalamnya ada unsur NPC yang mampu berkerumun.

Permasalahan yang timbul pada saat melihat kerumunan di Bandar udara adalah bagaimana kita menganalisa dari mana datangnya

setiap individu, bagaimana kerumunan terjadi. Tantangan pada analisa sekumpulan orang ini berada pada bagaimana sebuah kerumunan yang berbeda haluan bisa tetap sampai pada tujuan yang diinginkan masing- masing individu dalam kerumunan tersebut. Oleh karena itu maka kita harus memetakan semua pintu masuk, pintu keluar, layanan-layanan publik yang disediakan oleh Bandar udara tersebut seperti toilet, restoran, lounge, tempat tunggu dan tempat tempat yang lain yang rawan dikerumuni oleh individu-individu tersebut.

Setelah pemetaan tempat dilakukan maka akan dianalisa sumber dan tujuan individu-individu tersebut mulai saat individu - individu datang, kemudian berkerumun sampai individu-individu tersebut menyebar dan pergi sampai pada tujuan masing-masing.

Pada penelitian ini akan disimulasikan sebuah gerakan kerumunan di Bandar udara (*General airport*) menggunakan library RVO (*Reciprocal Velocity Obstacle*) library ini diperluas dengan menambah algoritma ORCA. Dimana ORCA ini memiliki 6 variabel yang mendefinisikan setiap individu yaitu r_a radius individu, a Posisi individu, V kecepatan terakhir individu, V_{max} kecepatan maximum individu, V_{pr} kecepatan yang diinginkan saat individu sudah sampai pada tujuan.

Sebagai pengujiannya kita akan membuat sumber-sumber individu dimana sumber-sumber datangnya individu tersebut akan membentuk sebuah kerumunan. Kerumunan-kerumunan yang sudah terbentuk akan dipertemukan dengan kerumunan yang lain dalam sebuah jalan terbuka. Pada setiap percobaan kerumunan-kerumunan ini akan diujikan dengan meningkatkan jumlah individunya

1.1 Rumusan Masalah

Keberadaan area seperti semua pintu masuk pada sebuah bandar udara merupakan tempat dan sumber munculnya individu-individu yang nantinya akan membentuk kerumunan di dalamnya.

Setelah individu berada dalam lingkungan bandar udara interaksi antar kerumunan akan dilihat dan disimulasikan berdasarkan tujuan masing-masing individu yang beradadalam kerumunan tersebut. Ada banyak kerumunan dengan tujuan yang berbeda-beda dalam simulasi inidibuat 3 kerumunan kerumunan 1, kerumunan 2, kerumunan 3 dimana setiap kerumunan terbentuk dari individu-individu dengan tujuan yang sama.

Pada waktu tertentu semua kerumunan yang berbeda tujuan dan berada pada satu koridor jalan berada pada area yang sama, akan bertemu dalam satu area, disinilah pengujian terhadap kerumunan akan dilakukan.

1.3 Batasan Masalah

Untuk membatasi masalah-masalah yang dibahas agar jelas maka permasalahan diatas diberi batasan-batasan masalah sebagai berikut:

1. Menentukan jumlah individu sebagai pembentuk kerumunan, individu ini akan

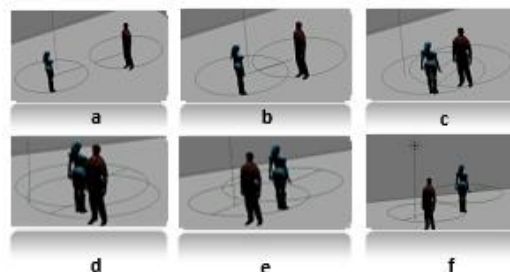
masuk dari pintu-pintu utama menuju tempat yang diinginkan.

2. Menentukan jumlah kerumunan-kerumunan yang dibentuk dari individu-individu yang berbeda haluan.
3. Menentukan sumber-sumber keluarnya individu yang masuk ke dalam area bandar udara.
4. Menentukan tujuan-tujuan kerumunan dan memetakan tempat - tempat yang memiliki kerawanan terjadinya kerumunan.
5. Memetakan tujuan akhir dari individu - individu yang berkerumun.

II. VELOCITY OBSTACLE

Velocity obstacle adalah pendekatan yang diterapkan pada kerumunan, pendekatan ini dilakukan dengan menitikberatkan kemampuan antar *agent* untuk bisa saling menghindari dari suatu tabrakan. *Velocity obstacle* mengadopsi kemampuan yang mengacu pada kecepatan individu antar *agent* sehingga mampu memilih kecepatan yang dibutuhkan agar bisa terhindar dari suatu tabrakan.

Pada sistem ini individu agent dengan yang lain tidak saling berkomunikasi, mereka hanya saling mengobservasi posisi masing-masing dari individu *agent* tersebut dan menentukan kondisi kecepatan agar bisa terhindar dari tabrakan dalam beberapa waktu saat individu agent saling berpapasan. Sebagai ilustrasinya bisa dilihat pada gambar 2.1b, pada posisi tersebut individu agent memiliki radius pada saat waktu belum bertemu atau berkonvolusi, kemudia posisi yang kedua adalah saat berpapasan (gambar 2.1c dan 2.1d) pada satuan waktu tertentu, dan saat individu agent telah berpapasan (gambar 2.1e dan 2.1f) dan tabrakan telah dihindarkan (gambar 2.1g).



Gambar 2. Collision Avoidance (kemampuan agent dalam menghindari tabrakan)

2.1 ORCA

ORCA adalah pengoptimalisasian agent agar mampu saling menghindari dari suatu tabrakan, ORCA bekerja berdasarkan program yang secara kontinyu terus menerus dilakukan seperti terlihat pada diagram berikut

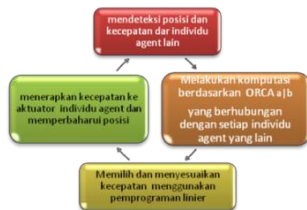


Diagram 2.1 Diagram looping sensing pada algoritma ORCA

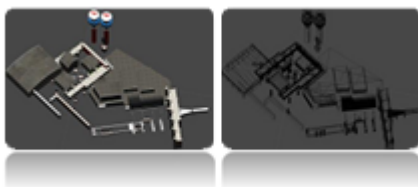
Setiap individu *agent* akan melakukan pendeteksian posisi dan kecepatan dari individu agent lain secara kontinu terhadap waktu ΔT seperti yang terlihat pada gambar diagram 2.1 di atas. Pada setiap iterasi diagram di atas maka individu agent, akan selalu mendapatkan informasi radius dan posisi terkini serta kecepatan terkini dari individu agent yang lain. Berdasarkan kemampuan ini maka individu agent akan mengambil tindakan dengan menyesuaikan kecepatannya berdasarkan kecepatan yang telah dipengaruhi oleh individu-individu agent lain yang berada disekitarnya berdasarkan rumus ORCA

$$R_{ij}^t = D(0, V_{ij}^{max}) \cap R_{ij}^t$$

Persamaan 1: Rumus Orca

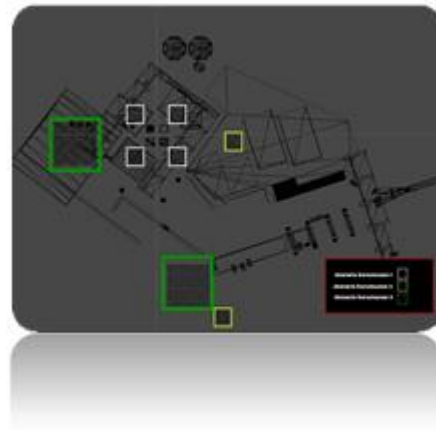
2.2 Bandar udara serta komponen

Bandar udara adalah suatu tempat dimana kerumunan mungkin terjadi baik itu sistematis ataupun tidak. Pada penelitian ini akan dititikberatkan pada bandara dengan area mengacu pada gambar sebagai berikut (**Gambar 2.2**).



Gambar 2.2 Area bandar udara

2.3 Pemetaan skenario area bandar udara yang rawan dikerumuni



Gambar 2.3 Titik daerah yang memiliki rawan kerumunan

Mengacu pada **gambar 2.3** skenario pertama akan dilakukan dengan memasang empat kerumunan pada titik titik yang ditandai warna putih, setiap kerumunan akan menuju ke tujuan dimana tujuan tersebut memiliki arah yang bertolak belakang dengan kerumunan *agent* lain, pada skenario ini kemampuan gerakan kerumunan akan diuji kemampuan daya hindar pada setiap kerumunan, namun demikian kerumunan tetap mampu menuju ke tujuan masing-masing.

Pada skenario ke dua yaitu kerumunan akan disebar pada 2 titik yang berwarna hijau, dan akan digunakan pintu yang sama untuk pintu keluar dan masuk dengan pembatas dinding yang berada di sebelah-sebelahnya, sehingga penyebaran kerumunan dibatasi dinding kanan dan dinding kiri. Pada skenario ini kemampuan gerakan kerumunan yang akan di uji kemampuan daya hindar pada setiap kerumunan, namun demikian kerumunan tetap mampu menuju ke tujuan masing-masing.

Pada skenario ke tiga yaitu kerumunan akan disebar pada 2 titik yang berwarna kuning dan akan digunakan satu celah sebagai tempat keluar masuk dari kerumunan tersebut. pada skenario ini kemampuan gerakan kerumunan akan diuji kemampuan daya hindar pada setiap kerumunan, namun demikian kerumunan tetap mampu menuju ke tujuan masing-masing.

III. PERANCANGAN SISTEM

3.1 Diagram Desain Sistem

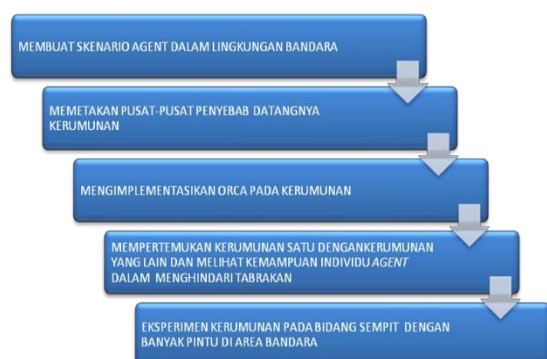


Diagram 3.1 Urutan pembuatan sistem

Pada perancangan penelitian ini akan dibagi lima bagian utama seperti urutan yang mengacu pada Diagram 3.1. Diagram desain sistem dibuat sebagai acuan urutan pada penelitian ini sehingga penelitian diharapkan tidak terlalu jauh fokus pembahasannya.

3.2 model *Human-alike* dan kerumunan



Gambar 3.2 Individu pembentuk kerumunan Searah jarum jam Agent1, agent2, agent3, agent4, agent5, agent6

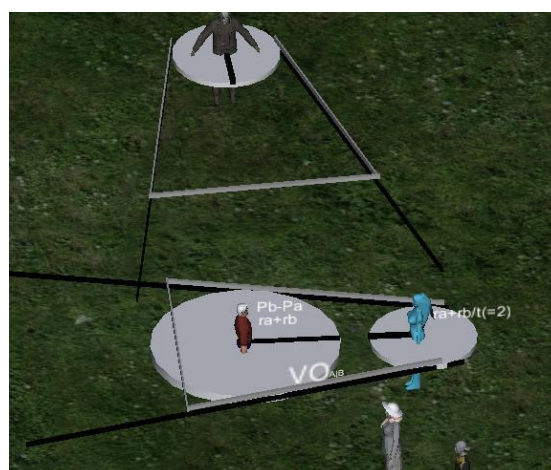
Untuk model yang akan digunakan sebagai pembentuk kerumunan digunakan enam model utama, identifikasi terhadap kerumunan akan dibedakan berdasarkan kecepatan dan model-model yang telah dibentuk seperti manusia seperti mengacu pada Gambar 3.2, untuk datanya bisa mengacu pada Tabel 3.2

Tabel 3.2 Tabel data pembentuk kerumunan

No	Nama	Amusi Klasifikasi Umur	Parameter Speed RVO	Parameter Untuk Time Horizon	V	Keterangan
1	Individu Agent 1	+10	0.1	0		Anak kecil
2	Individu Agent 2	+50	0.2	0		kakek
3	Individu Agent 3	+35	0.3	0		bapak
4	Individu Agent 4	+40	0.4	0		ibu
5	Individu Agent 5	+60	0.4	0		nenek
6	Individu Agent 6	+20	0.4	0		Pemuda

3.3 Implementasi ORCA pada kerumunan

Secara umum setiap *agent* mengadopsi algoritma ORCA, setiap agent melakukan sensing terhadap agen lainnya berdasarkan diagram yang terus menerus di *looping* sesuai dengan Gambar 3.3. ORCA akan melakukan *n-body collision avoidance* yang berarti panghindaran tabrakan diantara sejumlah (n) individu agent. Mengacu pada diagram 3.5a yang dilakukan secara terus menerus oleh setiap individu *agent* saat pergerakan kerumunan berjalan menuju tujuannya.



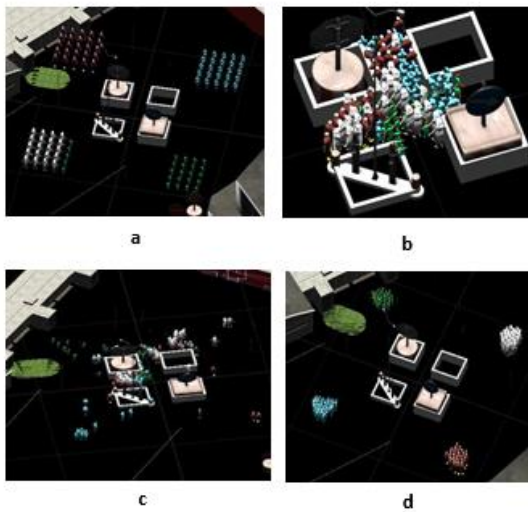
Gambar 3.3 Perhitungan ORCA untuk 2 agent

IV. IMPLEMENTASI SISTEM

4.1 Uji kerumunan pertama

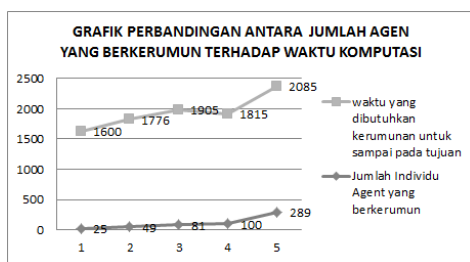
Pengujian tingkah laku pada 4 kerumunan disebar seperti mengacu pada gambar 4b, dan Dalam suatu lingkungan yang mengacu pada gambar 4a, secara global pengujian pada kerumunan dengan jumlah yang berbeda-beda, skenario untuk menguji kemampuan setiap agent dalam kerumunan adalah dengan mempertemukan

setiap kerumunan dengan kerumunan yang lain dalam satu ruang seperti pada gambar 4c.



Gambar 4.1 Gambar percobaan pertama

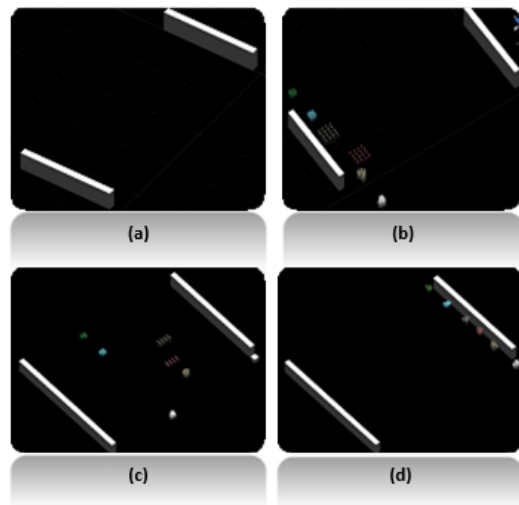
Dari percobaan yang telah dilakukan pada skenario pertama yang terdiri dari (25,49,81,100,289) agent yang masing-masing percobaan membentuk empat kerumunan ini didapatkan kesimpulan bahwa waktu komputasi collision free yang dibutuhkan oleh setiap agent dalam setiap kerumunanyang berbeda-beda jumlah ini, baik pada saat kerumunan berangkat dari titik asal kemudian bertemu dengan kerumunan yang lain dalam satu ruangan sampai kerumunan tersebut sampai pada tujuan akan semakin meningkat. Seperti yang tertera pada grafik 4.1



Grafik 4.1 Grafik perbandingan jumlah agen yang berkerumun terhadap waktu komputasi

4.2 Uji kecepatan kerumunan antar agent

Pada pengujian ini, semua model akan diuji kecepatannya terhadap jarak yang sama, seperti mengacu pada gambar 4.2.a.



Gambar 4.1 Gambar Pengujian kecepatan gerak antar kerumunan

Agents (Gambar 4.1 & 4.2) yang telah siap bergerak dari sumber kerumunan sampai ke tujuan akan diuji kecepatannya, hal ini untuk membedakan antara kecepatan beberapa agent yang satu terhadap agent yang lain.

Gambar 4.2 Agent Pembentuk kerumunan

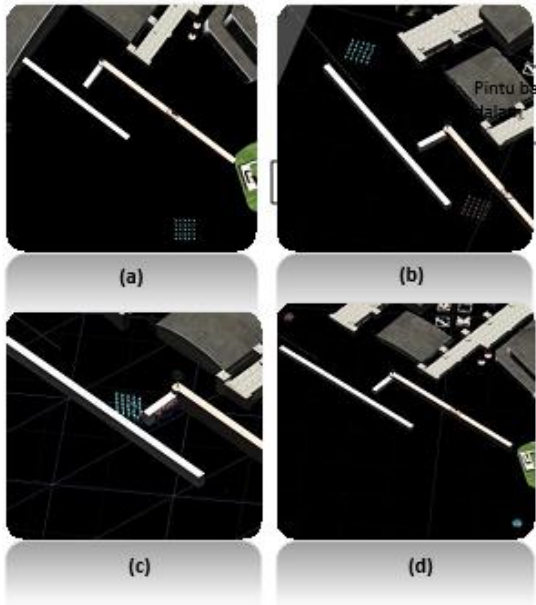


Seperti mengacu pada Gambar 4.1 agent akan bergerak dari satu titik ke Goal masing-masing dengan maxspeed yang berbeda-beda Kecepatan gerak kerumunan diukur berdasarkan waktu komputasi/frame terhadap goal yang dituju. Komponen yang diterapkan dalam kerumunan ini adalah nilai max speed yang diambil dari library RVO, dimana hasil ditunjukkan pada tabel 4.2.

Tabel 4.2 Tabel kecepatan gerak antar kerumunan

No	Nama	Anumri Klasifikasi Umur	Parameter Speed RVO	Koordinat sumber	Komputasi/frame	Koordinat tujuan
1	Agent 1/ Anak kecil	+10	0.5	-412,1274	808.5	-13.0f,1274.0f
2	Agent 2/ kakek	+50	1	-412,1129	414,75	-13.0f,1129.0f
3	Agent 3/ bapak	+35	0.8	-412,1073	516,75	-13.0f,1073.0f
4	Agent 4/ ibu	+40	0.5	-412,1200	808.5	-13.0f,1200.0f
5	Agent 5/ nenek	+60	0.5	-412,934	808.5	-13.0f,934.0f
6	Agent 6/ Pemuda	+20	0.9	-412,1020	452,75	-13.0f,1020.0f

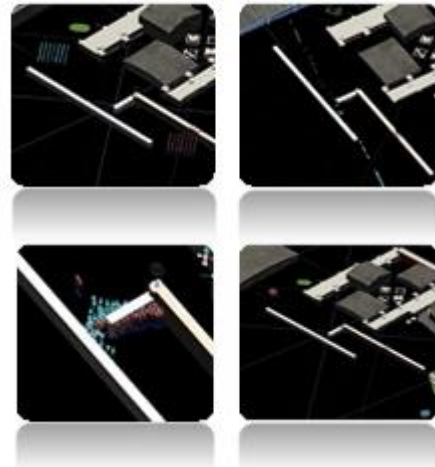
4.3 Uji kerumunan pada jalur koridor



Gambar 4.3 Pengujian kerumunan pada jalan koridor

Pada percobaan dengan skenario ini akan digunakan ruangan seperti yang telah dirancang sebelumnya, yaitu sebuah dinding yang dibatasi pada bagian kanan dan bagian kiri, hal ini dilakukan untuk mempersempit ruang gerak kerumunan, koridor ini di lengkapi dua pintu yaitu pintu dalam dan pintu luar seperti mengacu pada Gambar 4.3b (hal ini untuk menguji kerumunan agent saat melewati pintu yang lain).

Setiap kerumunan akan di skenario bergerak dari suatu titik menuju titik yang lain (Goal) dengan melihat kemampuan multi agentnya saat berpisah dengan kerumunanannya dan kemampuannya melewati jalan yang berbaeda menuju tujuannya, yaitu dengan melewati jalan dan pintu yang berbeda. Sebagai percobaan kita akan menggunakan 98 agent untuk pergerakannya.



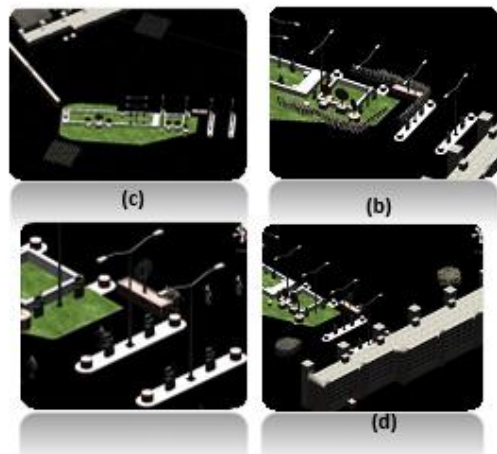
Gambar 4.4 Pengujian pada daerah berkoridor

Pada percobaan dengan 98 agent yang tersebar menjadi kerumunan ini agent bergerak dari sumber koordinat yang telah ditentukan seperti tertera pada Tabel 4.3b, kecenderungan agent saat berada pada bagian obstacle (dinding) adalah melakukan antrian seperti pada Gambar 4.4c, hal ini dikarenakan tidak adanya daerah bebas saat melakukan proses komputasi collision free, sehingga menunggu agent yang didepanya, dimana agent yang memiliki collision free harus bergerak terlebih dahulu agar bisa memberikan collision free pada agent yang sedang melakukan komputasi collision avoidance dibelakangnya, pada saat dimana collision free tidak ditemukan maka agent akan berada pada state kecepatan preferred velocity, sehingga hal ini akan membentuk antrian dibelakangnya hal ini seperti tampak pada Gambar 4.4c. Pada Tabel 4.1 dijelaskan hasil dari pergerakan bahwa komputasi dibuthkan sekitar 1776 step mulai dari saat agent berada pada state preposisi sampai agent berada pada tujuan.

Tabel 4.1 hasil dari pergerakan dua kerumunan yang terdiri dari 162 agent

No	Deskripsi	Vektor2 sumber	Vektor2 Goal	Waktu komputasi	G	Agent terjebak	Jumlah kerumunan
1	Agent3'apak	-360,-101	200,-670	1776	Y	3	49
2	Agent4'ibu	-24,-400	-374,-42	1776	Y		49

4.4 PERCOBAAN DENGAN 162 AGENT



Gambar 4.5 uji coba dengan 162 agent

Pada percobaan ini akan di buat skenario dengan kerumunan yang diletakkan pada area koordinat seperti pada gambar 4.5a, di mana a untuk menuju pada tujuan agent harus melalui celah seperti yang ditunjukkan pada panah di gambar 4.5a.

Dua kerumunan ini dibagi menjadi dua bagian dengan jumlah *agent* sebanyak 81 dan dibagi menjadi kerumunan *agent2* (*anakmuda*), dan *agent6* (*kakek*), kerumunan ini akan menuju goal masing- masing, dimana untuk menuju goal tersebut akan melalui celah seperti yang ditunjukkan pada panah di gambar 4.5a. Kerumunan yang mencari titik *collision free* namun tidak mendapatkan daerah itu akan membentuk antrian seperti yang terlihat pada gambar 4.5b. Setiap agent yang bergerak menuju goal, maka akan membuat *collision free* pada agent dibelakangnya, sehingga agent yang berada dibelakangnya akan memperbaharui posisi menuju goal, dan akan membuat kerumunan berkurang, komputasi yang terus menerus ini akan mengurai kerumunan. Dan pada akhirnya kerumunan sampai pada tujuan seperti pada gambar 4.5d, untuk waktu yang dibutuhkan dari preposisi sampai pada goal bisa dilihat pada tabel 4.3.3.

Tabel 4.3.3 hasil dari pergerakan dua kerumunan

No	Deskripsi	Vektor2 sumber	Vektor2 Goal	Waktu pergerakan	G	Agent terjebak	Jumlah kerumunan
1	Agent2/anak muda	215,-663	554,-86	1594	Y		81
2	Agent6/kakek	256,-44	639,-444		Y	2	81

V. KESIMPULAN

Dari hasil pengujian, dapat disimpulkan bahwa desain NPC kerumunan yang cerdas yang ditunjukkan dengan kemampuan hindar terhadap agen lain menggunakan metode RVO di area bandara udara telah berhasil dilakukan. Kesimpulan yang diperoleh dijelaskan secara detil sebagai berikut:

1. Implementasi RVO mampu diintegrasikan pada lingkungan software tiga dimensi, menggunakan software unity.
2. Agen mampu menuju goal dan mampu melewati halangan walaupun beberapa waktu bertemu dan bersilang arah dengan kerumunan yang lain.
3. Agen yang terjebak dan masuk kepada kerumunan lain mampu untuk keluar dari kerumunan tersebut dan kembali bergabung dengan kerumunannya.
4. Agen yang terjebak sangat lama dalam sebuah halangan disebabkan karena halangan tersebut berada pada jarak terjauh, dan agen tidak bisa berputar, sementara agen sudah dekat dengan goal.
5. Jumlah agen mempengaruhi waktu komputasi, semakin agen berkerumun dengan kerumunan yang lain maka komputasi semakin naik dan pergerakan semakin pelan.
6. Kerumunan mampu membentuk antrian saat bersilang arah dengan kerumunan yang lain pada pintu keluar yang sekaligus menjadi pintu masuk.
7. Kerumunan dengan RVO bisa diimplementasikan pada berbagai model agent dan membentuk kecepatan yang berbeda-beda, pada penelitian ini kami menggunakan 6 model *agent* (*agent 1.....6*) sebagai pembentuk elemen kerumunan
8. Setiap agen memiliki daya hindar saat bersilang arah dengan kerumunan yang lain dan berbaris membentuk sebuah antrian.
9. Dengan menggunakan *multiple* goal yang menjadi waypoint, mampu memanipulasi tujuan-tujuan kerumunan agent sehingga tapak lebih dinamis.
10. Algoritma ini bisa diimplementasikan pada kerumunan dengan model *Human alike*.

VI. DAFTAR PUSTAKA

- [1] Michael Negnevitsky, 2005, “Artificial Intelligence. A Guide to Intelligent Systems. Second Edition”, Pearson Education Limited, England, ISBN 0-321-20466-2.
- [2] BERG J. , GUY S. J. , LI N M. , MANOCHA D.: Reciprocal n-body collision avoidance. In International Symposium on Robotics Research (To Appear) (2009), Springer.
- [3] BERG J. , LI N M. , MANOCHA D.: Reciprocal velocity obstacles for realtime multi-agent navigation. Proc. of IEEE Conference on Robotics and Automation (2008),1928– 1935.
- [4] TREUILLE A., COOPER S., POPOVIC Z.: Continuum crowds. Proc. of ACM SIGGRAPH (2006), 1160– 1168.
- [5] REYNOLDS, C. W. 1987. Flocks, herds, and schools: A distributed behavioral model. In Computer Graphics (SIGGRAPH '87 Proceedings).
- [6] THALMANN, D., O’SULLIVAN, C., CIECHOMSKI , P., AND DOBBYN, S.2006. Populating Virtual Environments with Crowds. Eurographics 2006 TutorialNotes.
- [7] BAYAZIT, O. B., LIEN , J.-M., AND AMATO, N. M. 2002. Better group behaviors in complex environments with global roadmaps. Proc. 8th Intl. Conf. Artificial Life, 362–370.
- [8] D. Wilkie, J. VAN DEN BERG, D. MANOCHA. Generalized velocity obstacles. IEEE RSJ Int. Conf. Intell. Robot. Syst., 2009.

(10)

**SURVEY PADA METODE COLOR CONSTANCY UNTUK PERBAIKAN WARNA CITRA
DIPSTICK URINE**Ali S. Kholimi¹, R.V. Hari Ginardi²

Jurusan teknik Informatika Institut Teknologi Sepuluh Nopember Surabaya
dosen pengajar di Jurusan teknik Informatika Institut Teknologi Sepuluh Nopember Surabaya.
e-mail: kholimi@yahoo.com, hari@its.ac.id

ABSTRAK

Color Constancy merupakan metode yang digunakan untuk memperbaiki warna citra sehingga didapatkan warna citra yang sebenarnya. *Color constancy* banyak digunakan dalam perbaikan warna pada aplikasi *colorimetricanalysis*, yakni aplikasi pengenalan atau klasifikasi warna, sebelum proses klasifikasi warna dilakukan. Ada beberapa kelompok metode dalam *Color Constancy*, yakni metode statistik, metode Gamut dan metode pembelajaran. Metode statistik unggul dalam kecepatan, sedang metode pembelajaran unggul dalam performa hasil. Metode statistik juga lebih unggul dalam hasil dibandingkan dengan metode Gamut untuk citra dengan tingkat varian yang rendah. Dalam penelitian ini perbandingan beberapa metode statistik pada *Color Constancy* dilakukan untuk mengetahui metode mana yang berjalan paling baik pada citra dipstick urine. Metode-metode yang dibandingkan dalam penelitian ini adalah metode *White Patch*, *Gray World*, *Max-RGB*, *Shades of Grey*, dan *Grey Edge*. Perbandingan dilakukan dengan menghitung nilai RMSE citra hasil perbaikan dengan hasil pembacaan dipstick reader. Hasil yang didapatkan dari uji coba menunjukkan bahwa metode *white patch* lebih unggul hingga mencapai 25% terhadap hasil terbaik kedua dari metode-metode lain yang diuji.

Kata Kunci: *Color Constancy*, *Gray World*, *Grey Edge*, *Max-RGB*, *Shades of Grey*, *White Patch*

I. PENDAHULUAN

DIPSTICK adalah suatu alat yang digunakan untuk menguji kandungan suatu zat kimia dimana keberadaan zat kimia tersebut menunjukkan adanya kondisi tertentu. Dipstick yang dibahas pada penelitian di sini adalah dipstick yang digunakan untuk menganalisa penyakit diabetes.

Ada dua cara untuk membaca hasil dari sebuah dipstick, yakni dilakukan secara manual dan secara otomatis. Pembacaan dipstick secara manual dilakukan dengan cara mencocokkan warna reagent pada dipstick dengan tabel warna yang biasanya disertakan pada saat pembelian dipstick tersebut. Adapun pembacaan secara otomatis dilakukan dengan menggunakan sebuah perangkat yang disebut dengan dipstickreader. Dipstickreader ini bekerja seperti scanner dengan kemampuan mengenali warna.

Pembacaan dipstick secara manual mempunyai beberapa permasalahan, di antaranya rendahnya presisi dikarenakan tabel warna yang disertakan pada dipstick tidak menampilkan semua warna. Jika tabel warna mencantumkan semua

warna, tentu tabel warna tersebut akan terlalu besar untuk dibawa-bawa dan disertakan dalam penjualan. Adapun pembacaan secara otomatis dengan menggunakan dipstick reader mempunyai kekurangan dalam portabilitas perangkat. Selain perangkatnya cukup besar, perangkat dipstick reader juga hanya bisa digunakan untuk satu kegunaan saja, yakni membaca dipstick tersebut.

Dalam penelitian ini, kamera yang terdapat pada handphone digunakan dalam pengambilan citra dipstick. Pertimbangan digunakannya kamera yang terdapat pada handphone ini karena saat ini hampir semua orang memiliki handphone yang memiliki fasilitas kamera.

Penggunaan kamera pada handphone untuk menggantikan scanner dalam proses pengambilan citra dipstick ternyata memiliki beberapa kelemahan. Kamera dan scanner, pada dasarnya adalah perangkat yang sama-sama digunakan untuk mengakuisisi gambar dengan beberapa perbedaan.

John Capurso menjelaskan bahwa ada beberapa permasalahan dalam menggunakan kamera untuk menggantikan scanner ketika digunakan untuk mengakuisisi data, yakni permasalahan ketidak-stabilan pencahayaan, fokus, dan sudut pengambilan gambar.[1]

Dari beberapa permasalahan tersebut, dapat disimpulkan bahwa ada satu permasalahan utama dalam menggunakan kamera sebagai pengganti scanner dalam melakukan pembacaan pada dipstick, yakni bagaimana cara mendapatkan citra dari objek yang sama agar selalu tetap mendapatkan komposisi warna yang selalu sama walaupun berada dalam kondisi iluminasi yang berbeda?

Agar mendapatkan warna yang selalu stabil dalam proses pengambilan citra dari dipstick ini digunakan proses perhitungan color constancy. Perhitungan colorconstancy adalah sebuah metode pemrosesan citra yang ditujukan untuk memperkirakan warna sebenarnya dari sebuah citra yang diambil dari berbagai macam kemungkinan kondisi pencahayaan. [2][3][4] Hasil perkiraan ini akan memunculkan warna citra yang selalu sama terhadap sebuah objek dengan berbagai kondisi iluminasi. Konsistensi warna dalam pengambilan citra ini sangat diperlukan untuk ekstraksi fitur warna dalam aplikasi pengenalan warna. [3]

Aplikasi pengenalan warna banyak digunakan dalam permasalahan colorimetricanalysis. Penelitian dalam permasalahan colorimetricanalysis yang juga menggunakan metode colorconstancydi antaranya analisa kualitas air berdasarkan warna air [5], analisa bahan kimia dengan menggunakan mobile phone [6], dan pengenalan warna kulit seseorang [7].

Berdasarkan kompleksitasnya, Gijsenij[3] membagi algoritma pada color constancy menjadi 3 (tiga) jenis, yakni metode statistik, metode gamut, dan metode MachineLearning. Metode gamut sendiri sebenarnya adalah metode MachineLearning, namun karena memiliki kekhasan tersendiri, maka metode Gamut dan metode MachineLearning dipisahkan.

Kelebihan dari metode statistik adalah pada kompleksitasnya yang rendah, namun metode statistik ini mempunyai kelemahan pada foto yang memiliki varian tinggi. Metode Gamut dan Machine Learning bagus untuk foto dengan kompleksitas yang tinggi.

Dalam penelitian ini, perbandingan metode statistik pada color constancy diujikan pada data dipstick urine yang diambil dengan menggunakan kamera. Metode-metode dalam kelompok metode

statistik dipilih karena varian citra dari data dipstick yang rendah.

II. METODE STATISTIK DALAM COLOR CONSTANCY

a. White Patch

Metode White Patch Retinex digagas oleh Land yang juga penemu metode Retinex[3]. Metode White Patch Retinex adalah metode yang Uniform [8] dan bersifat statistik[3]. Berbeda dengan metode Retinex yang menyertakan patch warna putih dalam citra yang akan diambil, Metode White Patch Retinex mengansumsikan warna tertinggi pada semua channel warna RGB adalah sebagai warna putih. Kemudian masing-masing channel warna tersebut dilakukan skala ulang dengan warna putih maksimum tersebut sebagai warna maksimum untuk masing-masing channel.

Metode White Patch Color Constancy dilakukan dengan cara mendapatkan nilai maksimum dari tiap-tiap komponen warna RGB. Langkah pertama yang dilakukan adalah dengan cara mendapatkan histogram dari setiap komponen warna R, G, dan B. Setelah mendapatkan histogram dari setiap komponen warna RGB tersebut, langkah selanjutnya adalah mendapatkan warna maksimum dari masing-masing komponen warna RGB tersebut, atau warna paling kanan dari histogram, sebagaimana ditunjukkan oleh Persamaan (1), (2), dan (3).

$$\max R = \max(R) \quad (1)$$

$$\max G = \max(G) \quad (2)$$

$$\max B = \max(B) \quad (3)$$

Setelah mendapatkan nilai maksimum dari tiap-tiap komponen warna, langkah selanjutnya adalah menghitung rasio dari tiap-tiap komponen warna sebagaimana ditunjukkan oleh Persamaan (3), (4), dan (5). Nilai 255 digunakan karena nilai tersebut adalah nilai intensitas maksimum yang mungkin dimiliki dalam ruang warna RGB.

$$ratioR = \frac{255}{\max R} \quad (4)$$

$$ratioG = \frac{255}{\max G} \quad (5)$$

Patch, nilai rasio yang didapatkan tersebut kemudian dimasukkan ke dalam Persamaan (7), (8), dan (9) untuk mendapatkan hasil citra dari proses color constancy.

b. *Shades of Grey*

Shades of Grey merupakan bentuk umum dari metode Grey World dan White Patch.[3][9] Bentuk umum ini didapatkan dari pembuktian bahwa Grey World dan White Patch merupakan salah satu bentuk dari kerangka Minkowski. Di mana Grey World merupakan L1 Norm dari kerangka Minkowski dan Lmax Norm untuk White Patch. Shades of grey sendiri mengikuti kerangka Minkowski dengan nilai Norm L6. Persamaan umum yang berdasarkan pada bentuk normal Minkowski ini ditunjukkan oleh Persamaan (21).

$$\text{newI} = \left(\frac{\int (f(x))^p dx}{\int dx} \right) \quad (21)$$

c. *Grey Edge*

Grey World dan White Patch sendiri menggunakan nilai RGB Low Level dari citra. Van De Weijer[9] mengusulkan penggunaan nilai RGB high level yang merupakan turunan dari nilai RGB Low Level-nya, yang disebut dengan algoritma Grey Edge.

Grey Edge color constancy merupakan pengembangan dari metode Shade of Grey. Komputasi bentuk normal dari metode shades of Grey merupakan bentuk dari operasi rata-rata global tanpa mempertimbangkan korelasi lokal antar piksel. Padahal, korelasi lokal ini bisa digunakan untuk mengurangi kemungkinan kemunculan noise. Penghalusan dilakukan dengan menghaluskan citra terlebih dahulu dengan menggunakan filter Gaussian sebagaimana ditunjukkan oleh Persamaan (22).

$$f^\sigma = f \otimes G^\sigma \quad (22)$$

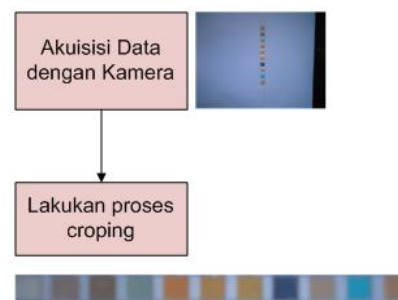
Citra yang dihasilkan kemudian menjadi masukan dari metode Shades of Grey sehingga persamaannya berubah menjadi Persamaan (23).

$$\text{newI} = \left(\frac{\int (f^\sigma(x))^p dx}{\int dx} \right) \quad (23)$$

III. SKENARIO UJI COBA

Data uji yang digunakan pada uji coba ini didapatkan dengan cara mengambil gambar Dipstick urine. Dipstick urine yang digunakan adalah dipstick urine yang telah dicelupkan ke urine pasien untuk memperbanyak variasi warna yang muncul. Pengambilan gambar dilakukan dengan menggunakan kamera. Selain dengan menggunakan kamera, data juga didapatkan dengan cara memindainya dengan dipstick scanner sebagai pembanding hasil dikarenakan dipstick scanner adalah alat yang umumnya digunakan untuk membaca dipstick ini.

Berbeda dengan hasil pemindaian dengan menggunakan dipstick scanner yang hanya memindai bagian dari dipstick saja, pengambilan citra dengan menggunakan kamera akan menyebabkan bagian lain yang bukan merupakan citra dipstick akan ikut terambil. Oleh sebab itu, proses cropping untuk mengambil area yang merupakan area citra dipstick harus dilakukan terlebih dahulu. Secara umum, proses pengambilan citra dipstick dengan menggunakan kamera ditunjukkan oleh Gambar 1.



Gambar 1. Proses pengambilan data dipstick urine dengan menggunakan kamera

Dipstick urine yang telah di-crop dari citra yang diambil dengan menggunakan kamera tadi untuk selanjutnya diproses dengan menggunakan 5 macam algoritma color constancy untuk melakukan perbaikan warna citra dari citra dipstick urine yang didapatkan dari kamera tersebut. Kelima algoritma yang diujikan yakni White Patch, Grey World, Max-RGB, Shades of Grey dan Grey-Edge. Proses perbaikan warna dengan menggunakan color constancy ini secara umum diperlihatkan oleh Gambar 2. Hasil dari perbaikan tersebut akan dianalisa dengan menghitung RMSE per-reagent pada masing-masing citra keluaran metode yang diujikan.



Gambar 2. Proses perbaikan warna dengan menggunakan color constancy

Perhitungan RMSE akan dilakukan dengan cara menghitung rata-rata warna per-reagent terlebih dahulu, sebagaimana ditunjukkan oleh Persamaan (24), dimana $I(n,C)$ adalah citra pada reagent ke- n dan ruang warna ke- C . Rata-rata warna per-reagent digunakan karena ukuran citra yang berbeda-beda. Untuk mendapatkan warna per-reagent, dilakukan proses cropping untuk setiap reagent. Hasil dari cropping ini yang akan diambil rata-ratanya.

$$\text{meanR}(n, C) = \text{mean}(I(n, C)) \quad (24)$$

Nilai rata-rata dari setiap reagent dari setiap komponen ruang warna ini nantinya dihitung nilai kuadratnya sebagaimana ditunjukkan oleh Persamaan (25). Hasil dari kuadrat ini kemudian dijumlahkan seluruhnya sebagaimana ditunjukkan oleh Persamaan (26) untuk kemudian dibagi jumlah total yang sama dengan jumlah reagent dikalikan jumlah ruang warna sebagaimana ditunjukkan oleh Persamaan (27). Persamaan (27) ini biasa disebut dengan Mean Square Error (MSE).

$$\text{powR}(n, C) = \text{mean}(n, C)^2 \quad (25)$$

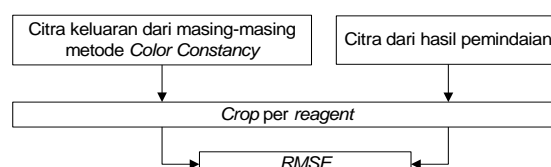
$$\text{sumR} = \sum_n \sum_c \text{powR}(n, c) \quad (26)$$

$$\text{MSE} = \frac{\text{sumR}}{|n|*|C|} \quad (27)$$

Hasil dari perhitungan MSE ini kemudian diakarkan untuk mendapatkan nilai dari RMSE sebagaimana ditunjukkan oleh Persamaan (28).

$$\text{RMSE} = \sqrt{\text{MSE}} \quad (28)$$

Nilai RMSE pada masing-masing metode dibandingkan dengan cara mencari nilai RMSE yang paling kecil di antara metode-metode tersebut. Nilai RMSE yang paling kecil menunjukkan kesalahan warna yang paling kecil. Secara umum, proses pencarian nilai RMSE yang digunakan sebagai nilai perbandingan ini ditunjukkan oleh Gambar 3.

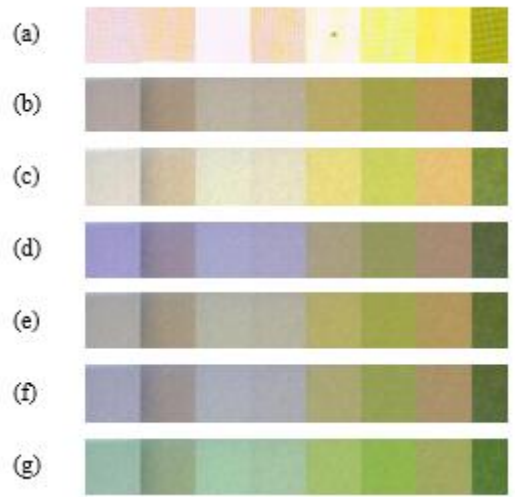


Gambar 3. Proses perhitungan RMSE

IV. HASIL UJI COBA

Gambar 4 memperlihatkan bahwa citra yang didapatkan dari pemrosesan citra menggunakan metode White Patch yang ditunjukkan oleh (c) memiliki kemiripan yang lebih besar dengan citra hasil scanner yang ditunjukkan oleh (a) dibandingkan dengan metode yang lain. Metode Max-RGB sendiri yang ditunjukkan oleh (e) tidak terlalu memperlihatkan perubahan dibandingkan dengan citra asalnya yang ditunjukkan oleh (b) walaupun bisa dikatakan lebih mendekati citra hasil pemindaian dibandingkan dengan yang lain.

Gambar 4 juga memperlihatkan bahwa citra hasil dari proses dengan menggunakan metode Grey World, yang ditunjukkan oleh (d), dan citra hasil proses dengan menggunakan metode Shades of Gray, yang ditunjukkan oleh (f), memiliki hasil yang jauh dari citra yang didapatkan dari pemindaian. Selain hasilnya juga jauh dari hasil yang didapatkan dari pemindaian, keduanya juga memperlihatkan hasil yang lebih menunjukkan semakin kuatnya dominasi intensitas warna biru. Berbeda dengan keluaran dari metode Grey Edge, yang ditunjukkan oleh (g), yang menunjukkan semakin kuatnya dominasi warna hijau.



Gambar 4. Citra Data 1 Keluaran dari (a) scanner, (b) kamera, (c) White Patch, (d) Grey World, (e) Max-RGB, (f) Shades of Grey, (g) Grey Edge

Hasil yang serupa juga ditunjukkan oleh Gambar 5, dimana kedua gambar ini, Gambar 4 dan Gambar 5, berasal dari dipstick yang berbeda sehingga memiliki warna asal yang berbeda juga. Meskipun berasal dari dipstick yang berbeda, ternyata Gambar 5 juga menunjukkan bahwa metode Max-RGB tidak terlalu memperlihatkan perubahan dibandingkan dengan citra asalnya. Selain itu, Gambar 5 juga menunjukkan semakin kuatnya dominasi warna biru pada keluaran dari metode Grey World dan Shades of Grey, serta semakin kuatnya dominasi warna hijau pada keluaran dari metode Grey Edge.



Gambar 5. Citra Data 2 Keluaran dari (a) scanner, (b) kamera, (c) White Patch, (d) Grey World, (e) Max-RGB, (f) Shades of Grey, (g) Grey Edge

Gambar 6 merupakan gambar yang berasal dari dipstick yang sama dengan Gambar 5 namun citra pada Gambar 6 ini diambil di waktu yang berbeda dengan faktor pencahayaan dan jarak pengambilan gambar yang berbeda. Gambar 5 dan Gambar 6 menunjukkan kekonsistenan hasil untuk metode yang sama pada citra berbeda dari dipstick yang sama. Hal ini menunjukkan bahwa perbedaan faktor pencahayaan dan jarak pengambilan gambar yang berbeda juga diproses pada masing-masing metode yang diujikan ini.

**TABEL 1
NILAI RATA-RATA RMSE DARI PENGUJIAN**

	RMSE	% Peningkatan Performa Terhadap Keluaran Kamera
Kamera	141,66	0 %
White Patch	105,37	25,61 %
Grey World	151,83	-7,18 %
Max-RGB	143,75	-1,48 %
Shades of Grey	147,95	-4,44 %
Grey Edge	135,86	4,09 %

Tabel 2, 3, dan 4 adalah tabel yang menunjukkan performa algoritma yang diimplementasikan dengan memperlihatkan nilai RMSE yang didapatkan pada setiap komponen warna dari ruang warna RGB. Tabel 2, 3, dan 4 menunjukkan bahwa peningkatan performa terbesar

dalam perbaikan warna ada pada warna merah dan warna hijau, secara berurutan. Adapun pada warna hijau, meskipun terjadi peningkatan performa, peningkatannya ini tidak terlalu besar.

TABEL 2
NILAI RATA-RATA RMSE DARI PENGUJIAN PADA
RUANG WARNA MERAH

	RMSE	% Peningkatan Performa Terhadap Keluaran Kamera
Kamera	75,81	0 %
White Patch	39,66	47,69 %
Grey World	94,62	-24,89 %
Max-RGB	81,20	-7,12 %
Shades of Grey	90,34	-19,18 %
Grey Edge	63,62	16,07 %

TABEL 3
NILAI RATA-RATA RMSE DARI PENGUJIAN PADA
RUANG WARNA HIJAU

	RMSE	% Peningkatan Performa Terhadap Keluaran Kamera
Kamera	77,82	0 %
White Patch	50,09	35,63 %
Grey World	84,91	-9,11 %
Max-RGB	79,30	-1,90 %
Shades of Grey	81,42	-4,63 %
Grey Edge	69,83	10,28 %

TABEL 4
NILAI RATA-RATA RMSE DARI PENGUJIAN PADA
RUANG WARNA BIRU

	RMSE	% Peningkatan Performa Terhadap Keluaran Kamera
Kamera	90,91	0
White Patch	83,79	7,82 %
Grey World	83,00	8,69 %
Max-RGB	88,22	2,95 %
Shades of Grey	84,24	7,34 %
Grey Edge	97,64	-7,41 %

V. KESIMPULAN

Paper ini melakukan sebuah survey terhadap beberapa metode statistik pada metode color constancy, yakni metode white patch, Grey World, Max-RGB, Shades of Grey, dan Grey Edge. Hasil perhitungan menunjukkan bahwa nilai RMSE citra keluaran dari metode White Patch adalah yang terbaik. Metode ini mampu menurunkan nilai RMSE hingga 25,61% dibandingkan dengan citra asalnya yang didapatkan dari kamera.

Meskipun begitu, metode baru untuk memperbaiki warna dipstick urine perlu diusulkan untuk meningkatkan performa, karena nilai RMSE yang baik untuk sebuah citra adalah nilai RMSE yang berada di bawah 30, sedangkan nilai RMSE yang didapatkan dari metode White Patch masih di atas angka 30.

VI. DAFTAR PUSTAKA

- [1] John Capurso. "Mobile Document Capture: Scanner vs. Phone Camera." Xerox White Paper. 2012.
- [2] Bianco, S., G. Ciocca, C. Cusano, dan R. Schettini. "Automatic color constancy algorithm selection and combination." *Journal of Pattern Recognition* Vol. 43. 2010.
- [3] Gijsenij, Arjan, Theo Gevers, dan Joost Van De Weijer. "Computational color constancy: Survey and experiments." *Image Processing, IEEE Transactions on* 20.9. 2011
- [4] Barnard, Kobus, Vlad Cardei, dan Brian Funt. "A comparison of computational color constancy algorithms. I: Methodology and experiments with synthesized data." *Image Processing, IEEE Transactions on* 11.9: 972-984. 2002.
- [5] Iqbal, Zafar, and Robert B. Bjorklund. "Colorimetric analysis of water and sand samples performed on a mobile phone." *Talanta* 84.4: 1118-1123. 2011
- [6] García, Antonio, dkk. "Mobile phone platform as portable chemical analyzer." *Sensors and Actuators B: Chemical* 156.1: 350-359. 2011.
- [7] Bianco, Simone, and Raimondo Schettini. "Color constancy using faces." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE.* 2012.
- [8] Ebner, Marc. *Color constancy*, Vol. 6. John Wiley & Sons Ltd, England. 2007.
- [9] Van De Weijer, Joost, Theo Gevers, and Arjan Gijsenij. 2007. "Edge-based color constancy." *Image Processing, IEEE Transactions on* 16.9: 2207-2214. 2007.

**PEDOMAN PENULISAN KARYA ILMIAH
JURNAL MelekIT
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK UNIVERSITAS WIJAYA KUSUMA SURABAYA**

Karya ilmiah merupakan hasil penelitian / perancangan / perencanaan seluruh masyarakat yang berkecimpung dibidang komunikasi dan informatika.

Susunan :

Terdiri dari judul, nama dan alamat penulis, abstrak, pendahuluan, metodologi, hasil dan pembahasan, kesimpulan dan saran, ucapan terima kasih (bila perlu), daftar pustaka

Format:

Ditulis dalam kertas A4 dua kolom, jenis huruf Times New Roman 10 *Regular* kecuali judul dengan huruf Times New Roman 11 *Bold*, satu spasi, maksimum 15 halaman dengan margin *left,top,right,bottom* (3,2.54,2.54,2.54).

ISI TULISAN:

Judul

Judul singkat dan jelas, ditulis menggunakan huruf besar dan cetak tebal/*bold*, maksimum 20 kata

Penulis

Nama penulis tanpa gelar, dosen pembimbing dapat menjadi penulis kedua dan seterusnya

Alamat

Merupakan alamat instansi atau badan hukum pelindung lainnya dan alamat email penulis.

Abstrak

Abstrak ditulis dalam satu paragraf dan maksimum 200 kata, berisi *highlight* hasil penelitian / perancangan / perencanaan diikuti dengan data kuantitatif yang terkait dengan judul serta hal yang perlu diungkapkan

Kata kunci

Kata kunci 3 sampai 5 kata/kalimat yang penting atau mewakili isi abstrak dan dapat digunakan sebagai kata penelusur, dan disusun berdasarkan urutan abjad

Pendahuluan

Pendahuluan berisi latar belakang dan dukungan kepustakaan yang diakhiri dengan tujuan penelitian, terdiri 3-5 paragraf.

Metodologi

Metodologi berisi tentang metode, rancangan, prosedur yang digunakan

Hasil dan Pembahasan

Pembahasan hasil penelitian disertai dukungan pustaka terkait. Tabel dan gambar diberi nomor dan judul, diawali dengan huruf besar tiap kata kecuali kata sambung

Kesimpulan dan saran

Kesimpulan ditarik dari hasil dan pembahasan dengan mengacu pada tujuan penelitian. Saran ditulis bila ada

Ucapan Terima Kasih (bila perlu)

Dapat dituliskan nama instansi atau perorangan yang berperan dalam pelaksanaan penelitian

Daftar Pustaka

Urutan pustaka disusun berdasarkan abjad. Judul ditulis huruf besar pada setiap awal kata, kecuali kata sambung dan kata depan. Nama majalah/jurnal/Buletin ditulis dengan singkatan baku. Urutan penulisan, sebagai berikut:

Pustaka jurnal:

Nama, Tahun. Judul. Nama Jurnal (cetak miring). Volume, Nomor: halaman.

Contoh:

Wu, X. and Prior, R.L., 2010. Identification and characterization *J Agric Food Chem* 53: 3101-3113

Giri, I.B.D., I.K. Sudarsana, dan N.M. Tutarani, 2008. Kuat Tekan dan Modulus Elastisitas Beton dengan Penambahan Styrofoam (Styrocon). *Jurnal Ilmiah Teknik Sipil*. Vol.12. No.1. hal. 75-85.

Pustaka buku:

Nama, Tahun. Judul (cetak miring). Penerbit. Kota

Contoh:

Jay, J.M., 2008. *Modern Food Microbiology*. Aspen Publs., Inc., Gaithersburg, Maryland

Dewobroto, W., 2005. *Aplikasi Rekayasa Konstruksi dengan Visual Basic 6.0 (Analisis dan Desain Penampang Beton Bertulang sesuai SNI 03-2847-2002)*. PT. Elex Media Komputindo. Jakarta.

Pustaka online:

<http://www.decaqon.com/appnotes/aw&safety.p df>. (21 Desember 2011, jam, menit, detik)

Artikel dikirim kepada:

Redaksi Jurnal MelekIT

Program Studi Teknik Informatika

Fakultas Teknik-UWKS

Jl. Dukuh Kupang XXV/54 Surabaya

Telp:031-5677577

Fax:031-5920741

Email:info@melekit.tif.ft-uwks.ac.id

Website:www.melekit.tif.ft-uwks.ac.id

Penulis bertanggung jawab terhadap isi karya ilmiah. Korespondensi mengenai karya ilmiah dialamatkan pada penulis dengan mencantumkan alamat email dan alamat lain yang mudah dihubungi dari salah satu penulis