Aplikasi Sistem Informasi Sumber Daya Manusia dengan Fitur DSS Menggunakan Metode Topsis pada PT. X

Radiant Victor Imbar, Benny Setiadi Hartanto
Jurusan S1 Sistem Informasi
Fakultas Teknologi Informasi, Universitas Kristen Maranatha
Jl. Prof. Drg. Suria Sumantri No. 65 Bandung 40164
email : radiant.vi@eng.maranatha.edu, ben.setiadi@yahoo.com

Abstract

PT.X is a company with large number of employees, but still use the manual method in processing employees data and also in selecting job applicants. Consequently, to obtain information, recruitment process and to prepare reports relating to the employees take a long time. The purpose of this research is to create an information system that can handle attendance, payroll and employees recruitment that is computerized and integrated, making it easier for the company in organizing employees data.The method done in this research is decriptive analysis with the case study approach. The data gathering is done through interview to companies. The application is made using C# programming language and SQL Server 2008 R2 base data. The result of this research shows the application of the information system helps the users to organize data more efficiently, accurately and reliable.

Key words: information system, attendance, payroll, recruitment, C#, SQL server 2008 R2.

I. Pendahuluan

Faktor sumber daya manusia merupakan faktor yang sangat penting dalam mendukung kinerja sebuah perusahaan. PT. X adalah sebuah perusahaan yang memiliki tingkat penyerapan tenaga kerja yang tinggi. Penyerapan tenaga kerja yang tinggi ini menjadi masalah tersendiri ketika posisi yang ditawarkan relatif banyak. Hal tersebut tentunya akan disertai oleh membanjirnya data pelamar kerja untuk memenuhi posisi yang ditawarkan.

Kondisi tersebut dapat menyebabkan produktivitas dari perusahaan terganggu dikarenakan kosongnya beberapa posisi. Disamping itu, proses untuk memilah-milah pegawai yang dilakukan oleh individu manusia sering kali mengalami kesalahan penempatan yang tidak sesuai dengan kualifikasi dari pelamar itu sendiri, dan proses ini memerlukan banyak biaya dalam pelaksanaannya dengan hasil yang kurang memuaskan.

Selain masalah dalam hal recruitment, perusahaan juga mempunyai masalah dalam hal penggajian dan absensi. Dikarenakan proses yang masih dilakukan secara manual, perusahaan memerlukan waktu yang cukup lama untuk menghitung gaji pegawai, dimana perusahaan harus menghitung dan mempertimbangkan berbagai faktor.

Oleh karena itu diperlukan perancangan dan desain sistem baru yang dapat menggantikan sistem lama dan memberikan keunggulan tersendiri dibandingkan dengan sistem sebelumnya. Perancangan sistem informasi ini diharapkan dapat menangani kendala atau permasalahan yang dihadapi oleh departemen/bagian *Human Resource* (HR). Perancangan sistem ini juga diharapkan dapat mendukung kinerja dari perusahaan secara menyeluruh.

Perancangan sistem ini meliputi sistem pengelolaan bank data pegawai, sistem absensi pegawai, sistem penggajian pegawai dan sistem recuitment tenaga kerja.

II. Rumusan Masalah

Dari latar belakang yang ada, dapat dirumuskan permasalahan sebagai berikut :

1. Bagaimana membuat aplikasi yang dapat mempermudah perusahaan dalam melakukan proses *recruitment* karyawan?
2. Bagaimana membuat aplikasi yang dapat menangani absensi dan penggajian karyawan?

III. Kajian Teori

III.1 Departemen Sumber Daya Manusia (SDM)

Departemen SDM bertanggung jawab terhadap pengelolaan sumber daya manusia dalam sebuah organisasi. Pengelolaan dari SDM yang ideal dalam organisasi memiliki delapan aspek yaitu (Noviyanto, 2009:4) :

1. Seleksi dan recruitment.
2. Pelatihan dan pengembangan (*training and development*).
3. Compensation and Benefit.
4. Manajemen Kinerja (*Performance Management*).
5. Perencanaan Karir (*Career Planning*). ke kelompok lainnya (horizontal carreer path).
6. Hubungan Karyawan (*Employee Relations*).
7. *Separation Management*.
8. *Personnel Administration* and HRIS.

III.2 *Decission Support System(DSS)*

DSS dirancang untuk memberikan informasi yang dapat membantu proses pengambilan keputusan dan kemampuan komunikasi untuk menjawab problema semi-terstruktur. Permasalahan yang terstruktur memiliki tiga elemen, yaitu intelegensia, rancangan dan pilihan yang secara jelas dapat teridentifikasi. Artinya, masih masih mungkin untuk menjelaskan algoritma atau pola pengambilan keputusan yang memungkinkan sebuah problema dapat dikenali dan dipahami,

alternatif solusi dapat dicari dan dievaluasi, serta solusi dapat dipilih. Permasalahan tidak terstruktur ,sebaliknya adalah suatu permasalahan dimana ketiga aspek di atas tidak dapt teridentifikasi sama sekali. Permasalahan semi-terstruktur adalah bilamana salah satu atau dua dari ketiga aspek di atas jelas (Suryadi K dan Ramdhani M.A, 1998:14).

Secara garis besar *decision support system* terbagi dalam tiga komponen besar, yaitu :

1. *Database*

Merupakan sistem database yang berisi kumpulan dari semua data bisnis yang dimiliki perusahaan, baik yang berasal dari data master maupun data transaksi perusahaan. Untuk keperluan DSS, diperlukan data yang relevan dengan permasalahan yang hendak dipecahkan.

2. *Model Base*

Merupakan model yang mempresentasikan permasalahan yang ada ke dalam bentuk kuantitatif, contohnya model matematika (AHP, Topsis, dll) sebagai dasar simulasi atau pengambilan keputusan, termasuk di dalamnya tujuan dari permasalahan (objective), komponen yang terkait, batasan-batasan yang ada (constraints) dan hal-hal terkait lainnya.

3. *Software System*

Komputer memiliki kemampuan untuk menyelesaikan persoalan (perhitungan matematika) yang sama dalam waktu relatif singkat. Dengan ini maka timbul interaksi antara sistem dan pemakai, terminal dan sistem perangkat lunak sebagai komponen-komponen dari *software system.*

Adapun tujuan dibuatnya DSS adalah :

1. Membantu manajer dalam pengambilan keputusan atas masalah semi-terstruktur.
2. Memberikan support bagi pertimbangan manajer dan bukan untuk menggantikan fungsi manajer.
3. Meningkatkan efektifitas keputusan yang diambil manajer lebih daripada perbaikan efisiensinya.

III.3 *Technique for Order Preferences by Similarity to Ideal Solution (*TOPSIS)

TOPSIS adalah salah satu metode pengambilan keputusan multikriteria (MCDS) yang pertama kali diperkenalkan oleh Yonn dan Hwang pada tahun 1981. Dengan ide dasarnya adalah bahwa alternatif yang dipilih memiliki jarak terdekat dengan solusi ideal dan yang terjauh dari solusi ideal negatif.

TOPSIS memperhatikan baik jarak ke solusi ideal positif maupun jarak ke solusi ideal negatif dengan mengambil hubungan kedekatan menuju solusi ideal. Dengan melakukan perbandingan pada keduanya, urutan pilihan dapat ditentukan.

Metode TOPSIS banyak digunakan pada beberapa model MADM (*Multi Attribute Decision Making)* dikarenakan metode ini memiliki beberapa keunggulan (Yoon dan Hwang,1981:23), yaitu :

1. Konsepnya sederhana dan mudah dipahami
2. Komputasinya efisien
3. Memiliki kemampuan untuk mengukur kinerja relatif dari alternatif-alternatif keputusan dalam bentuk matematis yang sederhana.

Berikut ini adalah matriks D yang memiliki m alternatif dengan n kriteria, dimana x_{ij} adalah pengukuran pilihan dari alternatif ke-i dalam hubungannya dengan kriteria ke-j.

$$D = \begin{pmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1m} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2m} \\ \vdots & & & & \\ x_{m1} & x_{m2} & x_{m3} & \dots & x_{mn} \end{pmatrix}$$

Langkah-langkah yang dilakukan dalam penyelesaian masalah menggunakan metode *TOPSIS* adalah sebagai berikut :

1. Normalisasi matrix keputusan

   Dalam prosedur ini, setiap nilai atribut diubah menjadi nilai yang *comparable*. Setiap normalisasi dari nilai r_{ij} dapat dilakukan dengan perhitungan :

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} {x_{ij}}^2}}, \quad \begin{matrix} untuk\ i = 1,2,3,\dots,m \\ j = 1,2,3,\dots,n \end{matrix}$$

   Sehingga didapat matriks R hasil normalisasi,

$$R = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \vdots & & & \\ r_{m1} & r_{m2} & \dots & r_{mn} \end{pmatrix}$$

   R adalah matriks yang telah di normalisasi, dimana m menyatakan alternatif, n menyatakan kriteria dan r_{ij} adalah normalisasi pengukuran pilihan dari alternatif ke-*i* dalam hubungannnya dengan kriteria ke-*j*.

2. Pembobotan pada matrix yang telah di normalisasi

Setelah di normalisasi, setiap kolom dari matriks R dikalikan dengan bobot-bobot (w_j) yang telah ditentukan oleh pembuat keputusan. Sehingga, *weighted normalized matrix* adalah V = RW

$$V = \begin{pmatrix} v_{11} & v_{12} & \dots & v_{1n} \\ v_{21} & v_{22} & \dots & v_{2n} \\ \vdots & & & \\ v_{m1} & v_{m2} & \dots & v_{mn} \end{pmatrix} = RW$$

$$= \begin{pmatrix} w_1 r_{11} & w_2 r_{12} & \dots & w_n r_{1n} \\ w_1 r_{21} & w_2 r_{22} & \dots & w_n r_{2n} \\ \vdots & & & \\ w_1 r_{m1} & w_2 r_{m2} & \dots & w_n r_{mn} \end{pmatrix}$$

Dimana W adalah

$$W = \begin{pmatrix} w_1 & 0 & 0 & \dots & 0 \\ 0 & w_2 & 0 & \dots & 0 \\ \vdots & & & & \\ 0 & 0 & 0 & \dots & w_n \end{pmatrix}$$

3. Menentukan Solusi ideal positif dan Solusi ideal negatif

Solusi ideal dinotasikan dengan A^+ dan solusi ideal negatif dinotasikan dengan A^-, sebagai berikut :

$$A^+ = \{(\max\ v_{ij} \mid j \in J), (\min v_{ij} \mid j\ \in J'), i = 1,23, \dots, m\}$$
$$= \{{v_1}^+, {v_2}^+, \dots, {v_n}^+\}$$
$$A^- = \{(\min\ v_{ij} \mid j \in J), (\max v_{ij} \mid j\ \in J'), i = 1,23, \dots, m\}$$
$$= \{{v_1}^-, {v_2}^-, \dots, {v_n}^-\}$$

Dimana :
$J = \{j = 1,2,3, \dots, n$ dan j berhubungan dengan *benefit criteria*$\}$
$J' = \{j = 1,2,3, \dots, n$ dan j berhubungan dengan *cost criteria*$\}$

Pembangunan A^+ dan A^- adalah untuk mewakili alternatif yang *most preferable* ke solusi ideal dan yang *least preferable* secara berurutan.

4. Menghitung *separation measure*

Separation measure ini merupakan pengukuran jarak dari suatu alternatif ke solusi ideal dan solusi ideal negatif. Perhitungan matematisnya adalah sebagai berikut :

a. Solusi ideal positif

$${S_i}^+ = \sqrt{\sum_{i=1}^{n} (v_{ij} - {v_j}^+)^2}\text{, untuk } i = 1,2,3, \dots, m$$

b. Solusi ideal negatif

$S_i^- = \sqrt{\sum_{i=1}^{n}(v_{ij} - v_j^-)^2}$, untuk $i = 1,2,3, \ldots, m$

5. Menghitung kedekatan relatif dengan solusi ideal

Kedekatan relatif dari alternatif A_i dekat dengan solusi ideal A^+ direpresentasikan dengan :

$C_i^+ = \frac{S_i^-}{S_i^+ + S_i^-}$, dimana $0 < C_i^+ < 1$ dan $i = 1,2,3,\ldots,m$.

Dikatakan alternatif A_i dekat dengan solusi ideal apabila C_i^+ mendekati 1. Jadi, $C_i^+ = 1$, jika $A_i = A^+$, dan $C_i^- = 0$, jika $A_i = A^-$.

6. Mengurutkan pilihan

Pilihan akan diurutkan berdasarkan pada nilai C_i^+, sehingga alternatif dengan jarak terpendek dengan solusi ideal adalah alternatif yang terbaik.

III.4 *Entity Relationship Diagram*

Entity Relationship Diagram (ERD) adalah suatu penyajian data menggunakan Entity dan Relationship. ERD merupakan peralatan pembuatan model data yang paling fleksibel dan dapat diadaptasi untuk berbagai pendekatan yang mungkin diikuti perusahaan dalam pengembangan sistem. ERD ini menggambarkan relasi atau hubungan antar entitas yang ada.

III.5 *Flowchart*

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urut-urutan prosedur dari suatu program. *Flowchart* menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian (Febriani, 2007:1). Flowchart biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.

III.6 Unified Modelling Language (UML)

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. *Unified Modeling Language (UML)* adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut. UML mulai diperkenalkan oleh Object Management Group, sebuah organisasi yang telah mengembangkan

model, teknologi, dan standar OOP sejak tahun 1980-an. Sekarang UML sudah mulai banyak digunakan oleh para praktisi OOP. UML merupakan dasar bagi perangkat (tool) desain berorientasi objek dari IBM.
Empat macam diagram yang paling sering digunakan dalam pembangunan aplikasi berorientasi object, yaitu *Use case diagram, sequence diagram, collaboration diagram dan Class Diagram.*

IV. Analisis Dan Rancangan Sistem

Analisis rancangan sistem aplikasi ini dibagi atas 2 bagian yaitu analisis tujuan dan analisis proses aplikasi.

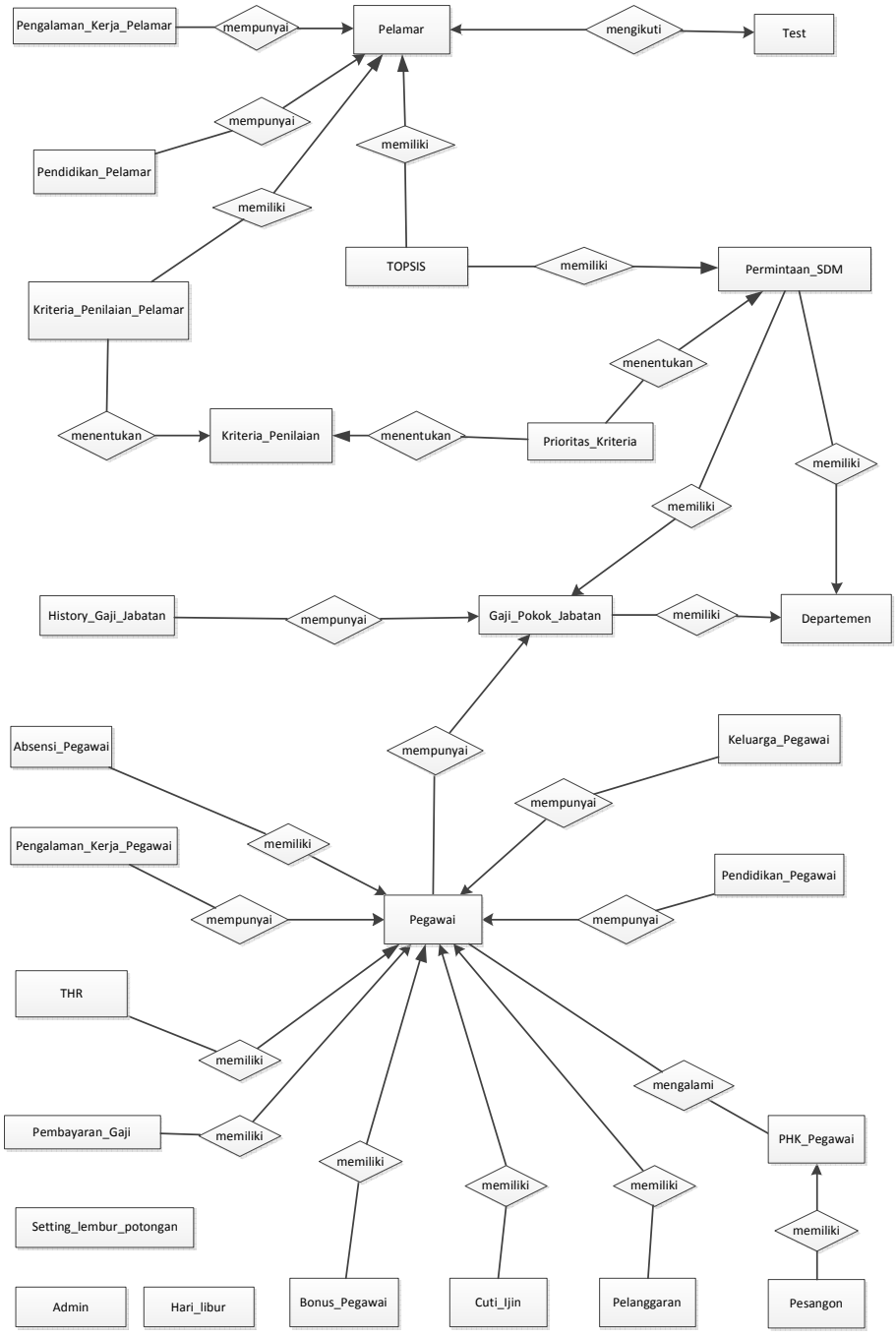
IV.1 Analisis Tujuan

Tujuan dari dirancangnya aplikasi ini adalah sebagai berikut :

1. Merancang database dan membangun sebuah sistem yang dapat mengelola data-data yang berhubungan dengan pegawai.
2. Memudahkan tugas dari manajemen SDM dalam mengelola dan mengakses data dengan cepat, tepat, dan akurat.
3. Meminimalkan kerusakan dan kehilangan data-data penting.
4. Merancang sistem absensi pegawai yang diharapkan dapat mengurangi kecurangan dan memudahkan dalam mengelola data-data yang berhubungan dengan kehadiran pegawai.
5. Merancang sistem penggajian dan sistem recruitment pegawai.

IV.2 Rancangan Enitity Relationship Diagram (ERD)

Perancangan ERD mencakup perencanaan tabel-tabel yang akan digunakan dalam database sehingga database efektif, efisien serta mengurangi redudansi data. Hal tersebut dapat tercapai dengan mendesain tabel-tabel dengan panjang dan tipe field yang sesuai dan efisien. Penamaan atribut yang jelas serta penentuan primary key yang tepat dapat mempermudah relasi antar tabel. Rancangan entity relationship diagram dapat dilihat pada gambar 1.



Gambar 1 ***Entity Relationship Diagram***

Berikut ini adalah penjelasan atribut-atribut yang digunakan dalam ERD :

1. **Pegawai** :
   ID_pegawai, ID_jabatan Nama_pegawai, Tgl_lahir, Tempat_lahir, Jenis_kelamin, Alamat, Kota, Tgl_bekerja, Agama, Status_nikah,

Status_pegawai, No_telp, Tanggungan_anak, Tunjangan, Gol_darah, Foto, *Password*_pegawai

2. **Pengalaman_kerja_pegawai**
   No, ID_pegawai, Nama_instansi, Jenis_instansi, Alamat_instansi, Telp_instansi, Tgl_masuk, Tgl_keluar, Alasan_berhenti, Lama_kerja

3. **Pendidikan_pegawai**
   No, ID_pegawai, Nama_institusi, Alamat_institusi, Telp_institusi, Tgl_mulai, Tgl_selesai, Jenjang, Jurusan, Ket_pendidikan

4. **Keluarga_pegawai**
   No, ID_pegawai, Nama_kel, Jenis_kelamin_kel, Hubungan, Alamat_kel, Tgl_lahir_kel, Pekerjaan, Jenjang_pendidikan_kel, Keterangan_keluarga

5. **Departemen**
   ID_Departemen, Nama_departemen

6. **Gaji_pokok_jabatan**
   ID_jabatan, ID_departemen, Nama_jabatan, Gaji_pokok, Uang_makan, Uang_transport, Total_gaji_jabatan

7. **History_gaji_jabatan**
   No, ID_jabatan, Total_gaji_update, Tgl_update

8. **Setting_lembur_potongan**
   No, Nama_setting, Jenis_setting, Nilai

9. **Pelanggaran**
   No_pelanggaran, ID_pegawai, Tgl_pelanggaran, Ket_pelanggaran

10. **PHK_pegawai**
    No_PHK, ID_pegawai, Tgl_PHK, Ket_PHK

11. **Pesangon**
    No_pesangon, ID_pegawai, No_PHK, Tgl_pesangon, Gaji_terakhir, Nilai_pesangon

12. **Admin**
    Username, *Password*, Status_aktif

13. **Hari_libur**
    No, Tgl_awal, Tgl_akhir, Ket_libur

14. **Absensi_pegawai**
    No_absensi, Tgl_absensi, ID_pegawai, Status_kehadiran, Jam_masuk, Jam_keluar, Lembur1, Lembur2, Lembur_libur, Foto_masuk, Foto_keluar,

Foto_lembur, Keterangan

15. **Cuti_ijin**
    No, ID_pegawai, Jenis, Keterangan, Tgl_mulai, Tgl_berakhir, Jumlah_hari, Tgl_pengesahan

16. **Bonus_pegawai**
    No_bonus, ID_pegawai, Ket_bonus, Tgl_bonus, Nilai_bonus

17. **Pembayaran_gaji**
    No_pembayaran, ID_pegawai, Bulan, Tahun, Gaji_jabatan, Upah_lembur1, Upah_lembur2, Upah_lembur_libur, Bonus, Tunjangan, Total_gaji, Tgl_pembayaran, Potongan

18. **THR**
    No_THR, ID_pegawai, Tahun, Jumlah_THR, Tgl_bayar

19. **Permintaan_SDM**
    No_permintaan, ID_jabatan, Jumlah, Ket_permintaan, Status_permintaan, Tgl_bekerja, Tgl_permintaan, ID_departemen

20. **Pelamar**
    No_form, Nama_pelamar, Tgl_lahir, Tempat_lahir, Jenis_kelamin, Alamat_pelamar, Kota, Agama, Status_nikah, Minat_jabatan, No_telp, Gol_darah

21. **Pendidikan_pelamar**
    No, No_form, Nama_institusi, Alamat_institusi, Telp_institusi, Tgl_mulai, Tgl_selesai, Jenjang, Jurusan, Ket_pendidikan

22. **Pengalaman_kerja_pelamar**
    No, No_form, Nama_instansi, Jenis_instansi, Alamat_instansi, Telp_instansi, Tgl_masuk, Tgl_keluar, Alasan_berhenti, Lama_kerja

23. **Test**
    No_test, No_form, Tgl_test, Minat, Bakat, IQ, Ket_test

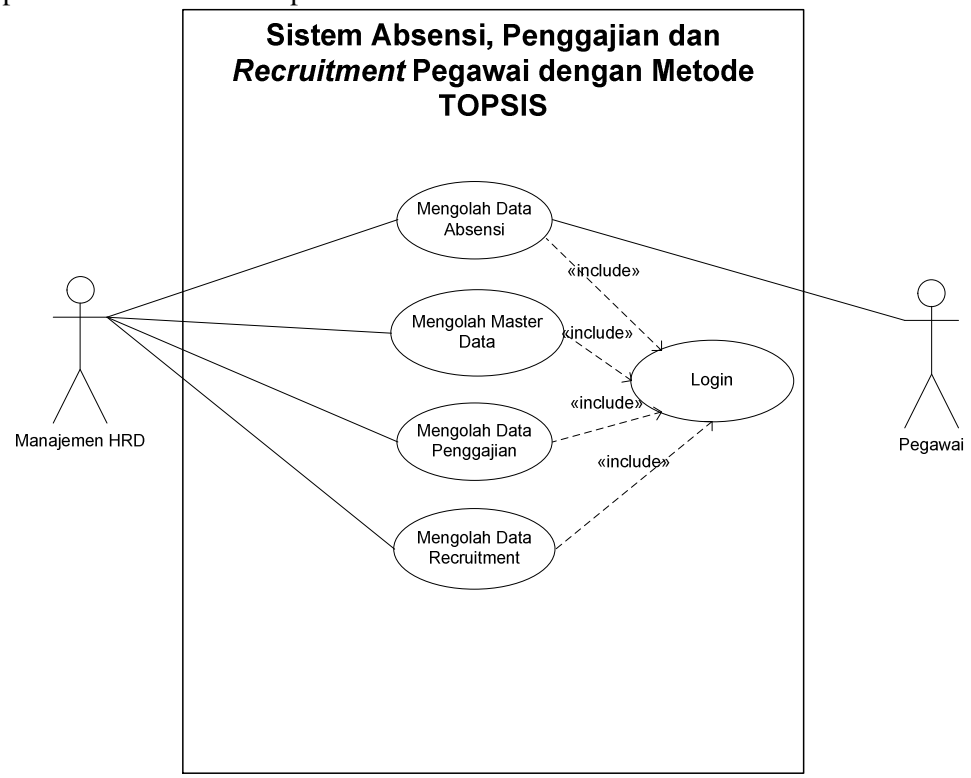
24. **Kriteria_penilaian**
    ID_kriteria, Nama_kriteria

25. **Prioritas_kriteria**
    No, No_permintaan, ID_kriteria, Bobot_prioritas
26. **Kriteria_penilaian_pelamar**
    No, No_form, ID_kriteria, Nilai

27. **TOPSIS**
    No, No_form, No_permintaan, Solusi_ideal, Status_seleksi

IV.3 Rancangan Use Case Diagram

Use case diagram adalah form tertinggi tentang suatu sistem yang merupakan gambaran pada pihak manajemen, customer, atau orang lain yang terlibat langsung pada sistem tersebut tetapi tidak secara mendetail.

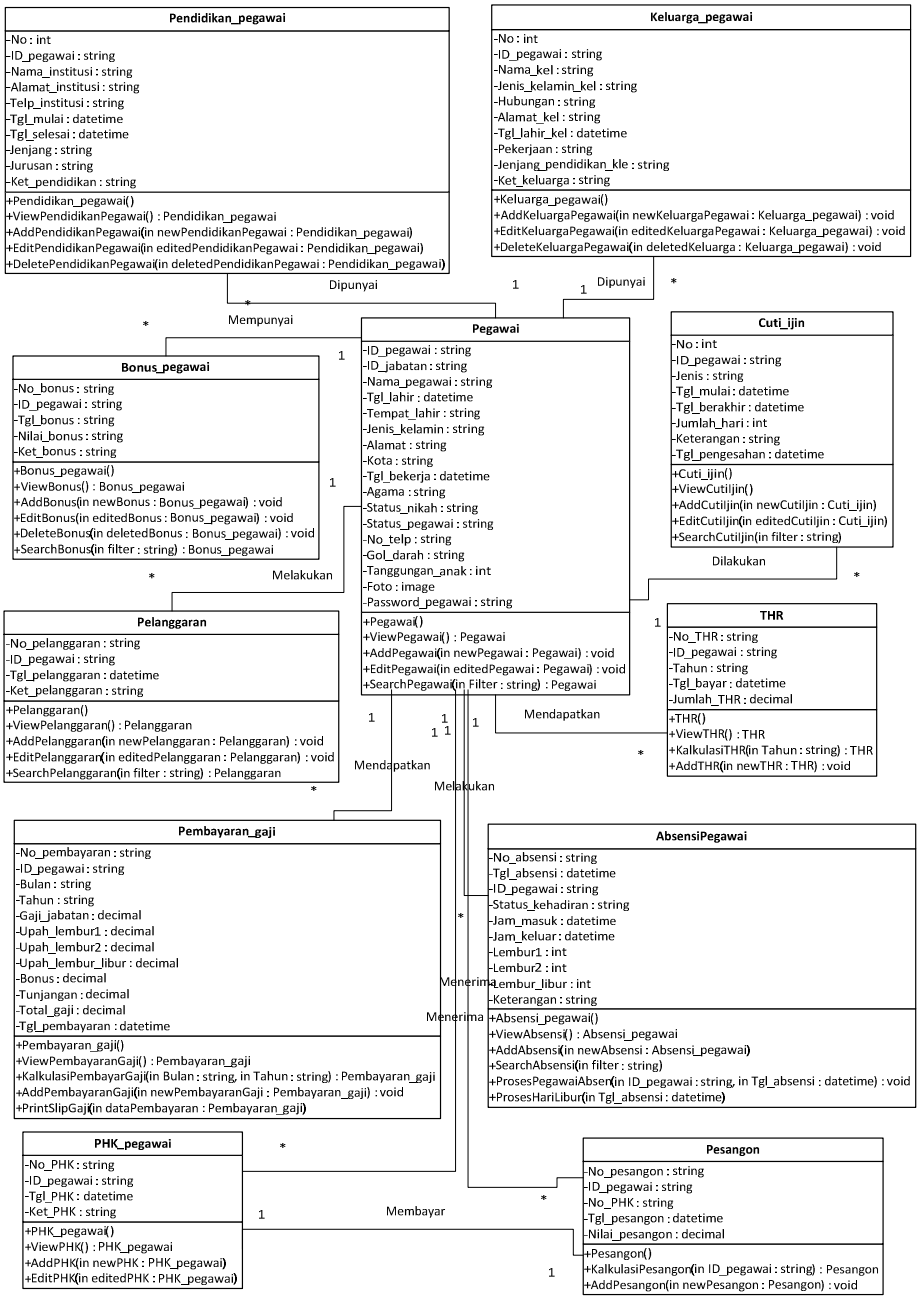


Gambar 2 *Use Case* Level 1

Pada Gambar 2 dapat dilihat *use case* level 1 terdapat 4 *cases*, yaitu mengolah data absensi, mengolah master data, mengolah data penggajian dan mengolah data *recruitment* yang dapat dilakukan oleh pihak Manajemen HRD, sedangkan pegawai hanya dapat melakukan input data absensi.

IV.4 Rancangan Class Diagram

Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. Berikut ini adalah gambar 3 yaitu class diagram aplikasi :

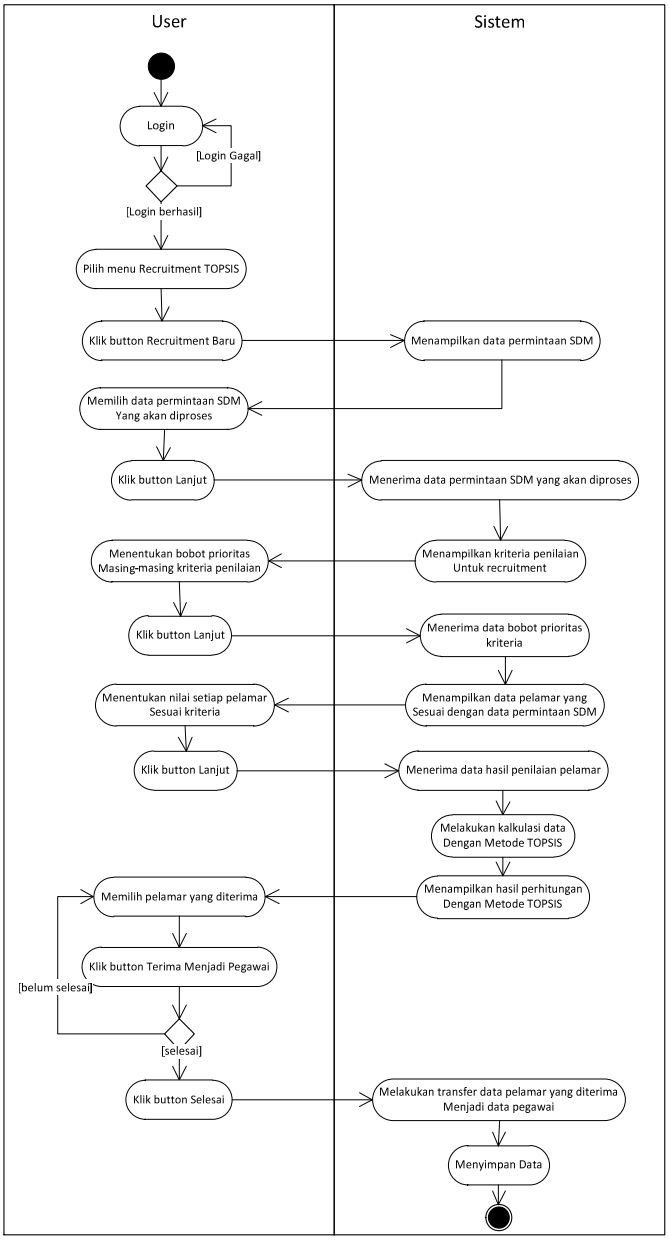


Gambar 3 *Class Diagram*

IV.5 Rancangan Activity Diagram

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin

terjadi, dan bagaimana mereka berakhir. Berikut ini adalah gambar 4 yaitu *activity diagram add recruitment* TOPSIS:

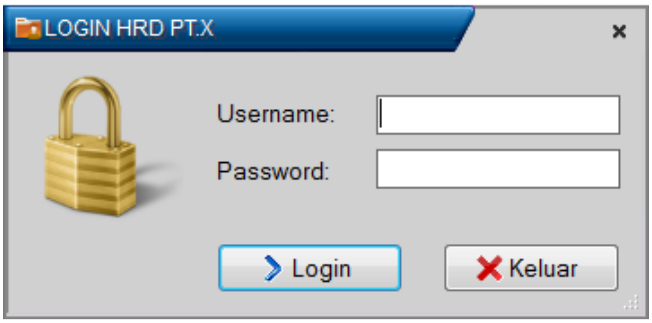


Gambar 4 *Activity Diagram Add Recruitment* TOPSIS

V. Hasil Tercapai

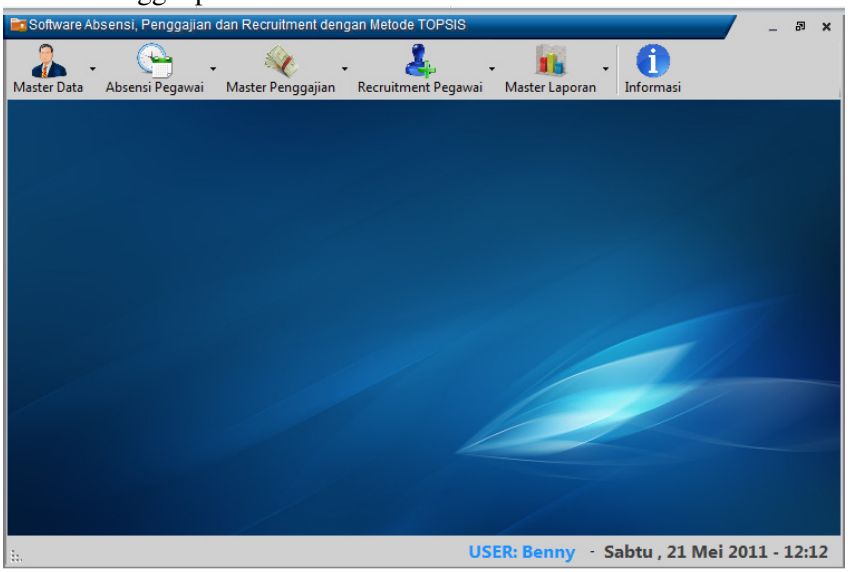
Gambar 5 adalah Tampilan login dimana setiap *user* harus mengisi *username* dan *password* untuk dapat mengakses program. Untuk menambah keamanan, semua

password yang dimasukkan *user* dienkripsi dengan algoritma *hash* MD5. Bila *user* berhasil login, maka *user* akan masuk ke menu utama.



Gambar 5 Tampilan Login

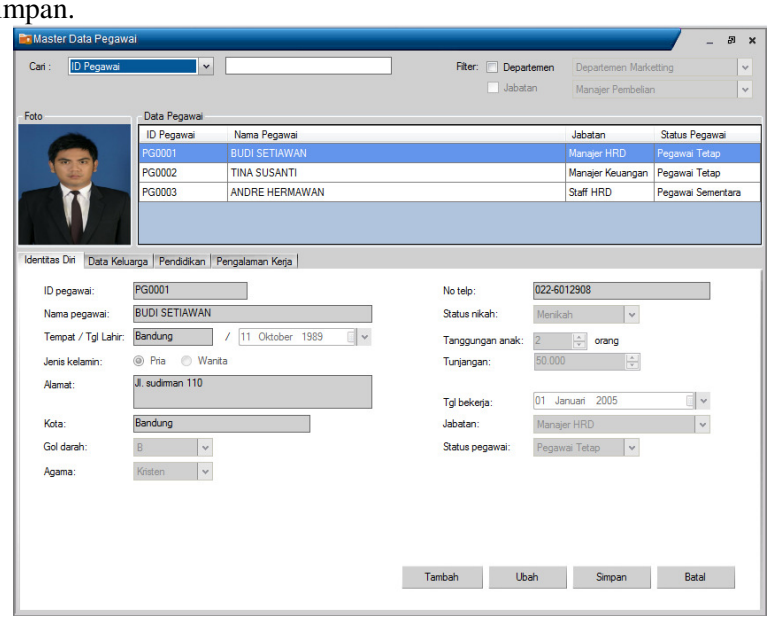
Setelah *user* berhasil login, maka *user* akan masuk ke menu utama. Gambar 6 adalah tampilan menu Master Data dimana pada menu Master Data terdapat menu Master Pegawai, Master Departemen, Master Gaji Pokok dan Jabatan, Pelanggaran, PHK dan Pesangon, Hari Libur dan Data Admin. Pada menu Absensi Pegawai terdapat menu Absensi Masuk Pegawai, Absensi Pulang Pegawai, Absensi Lembur Pegawai, Data Cuti Ijin dan Data Absensi Pegawai. Pada menu Master Penggajian terdapat menu Pembayaran Gaji, Bonus Pegawai dan Pembayaran THR. Pada menu *Recruitment* Pegawai terdapat menu Permintaan SDM, Data Pelamar, Data Kriteria, Data Hasil Test dan *Recruitment* TOPSIS. Menu Master laporan berisi laporan-laporan, seperti Laporan Data Pegawai, Laporan Pembayaran Gaji dan sebagainya. Pada bagian kanan bawah terdapat status *user* yang sedang aktif dan juga jam serta tanggal pada saat itu.



Gambar 6 Tampilan Menu Utama

Gambar 7 adalah gambar menu master pegawai. Menu master pegawai berisi data lengkap mengenai pegawai yang bekerja di perusahaan. Menu ini memiliki 4 tab di bagian bawah, yaitu tab Identitas, data keluarga, pendidikan dan pengalaman kerja. Pada bagian atas terdapat filter departemen dan jabatan yang dapat digunakan untuk menampilkan data pegawai pada departemen atau jabatan tertentu.
Untuk melakukan pencarian pegawai,*user* dapat memilih kriteria pencarian lalu memasukkan data yang ingin dicari. Untuk menambah data *user* dapat menekan *button* Tambah pada tab Identitas. Foto pegawai dapat diambil dengan 2 cara, yaitu mengambil foto pegawai dengan menggunakan *webcam* atau dengan memilih *file* gambar yang ada di komputer. Setelah semua data dimasukkan, *user* dapat menekan tombol simpan.

Untuk mengubah data, *user* dapat memilih data pegawai yang akan diubah, kemudian menekan *button* Ubah. Setelah semua data diubah *user* dapat menekan *button* Simpan.

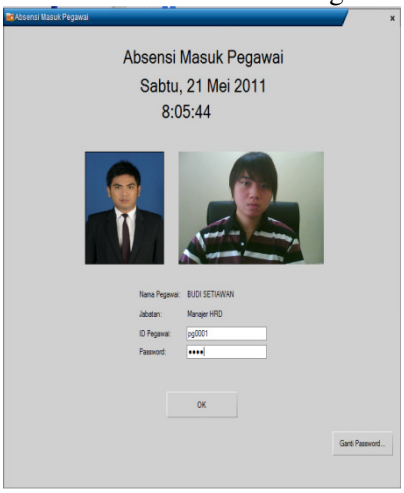


Gambar 7 Tampilan Master Pegawai

Pada menu input absensi (dapat dilihat pada gambar 8), pegawai dapat melakukan absensi masuk. Pegawai harus memasukkan ID Pegawai dan password untuk melakukan absensi. Foto, Nama serta jabatan pegawai akan muncul setelah pegawai memasukkan ID Pegawai. Untuk menambah keamanan dan keakuratan data absensi, foto pegawai saat melakan absensi juga diambil melalui webcam.
Pegawai yang ingin mengubah password dapat menekan button Ganti Password setelah memasukkan ID Pegawai. Setelah itu pegawai akan diminta untuk memasukkan password lama dan password baru.

Untuk mencegah form absensi ditutup secara tidak sengaja oleh pegawai, user harus memasukkan password saat menutup form ini.

Berikut ini adalah tampilan Menu Absensi Masuk Pegawai:

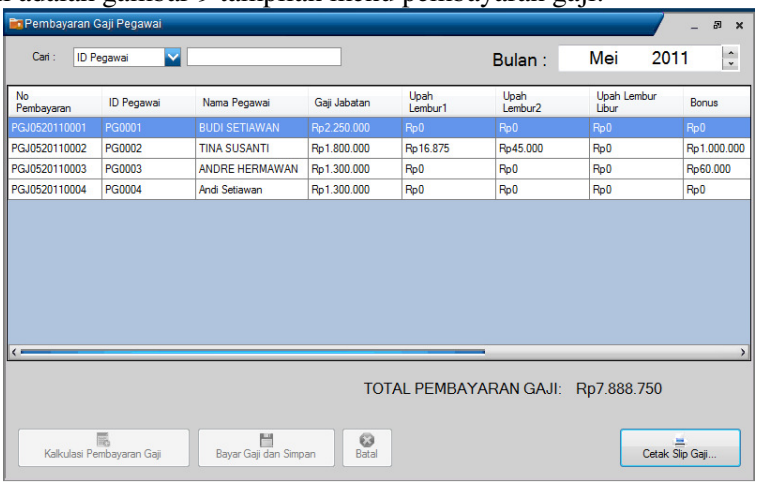


Gambar 8 Tampilan Absensi Masuk Pegawai

Untuk melakukan perhitungan atau kalkulasi pembayaran gaji, *user* dapat menekan *button* Kalkulasi Pembayaran Gaji. Setelah itu, data pembayaran gaji yang harus dibayar akan ditampilkan. Untuk menyimpan data, *user* dapat memilih *button* Bayar Gaji dan Simpan.

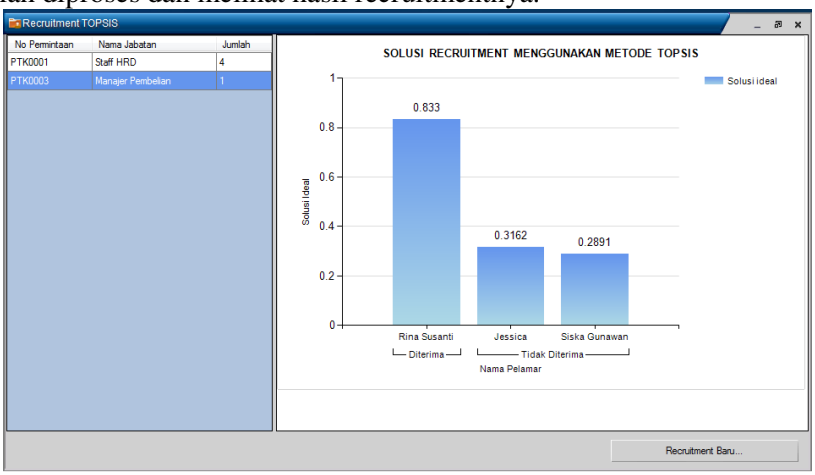
Untuk mencetak slip gaji, *user* dapat menekan *button* Cetak Slip Gaji. Setelah tampilan slip gaji muncul, *user* dapat menekan tombol *Print*.

Berikut ini adalah gambar 9 tampilan menu pembayaran gaji:



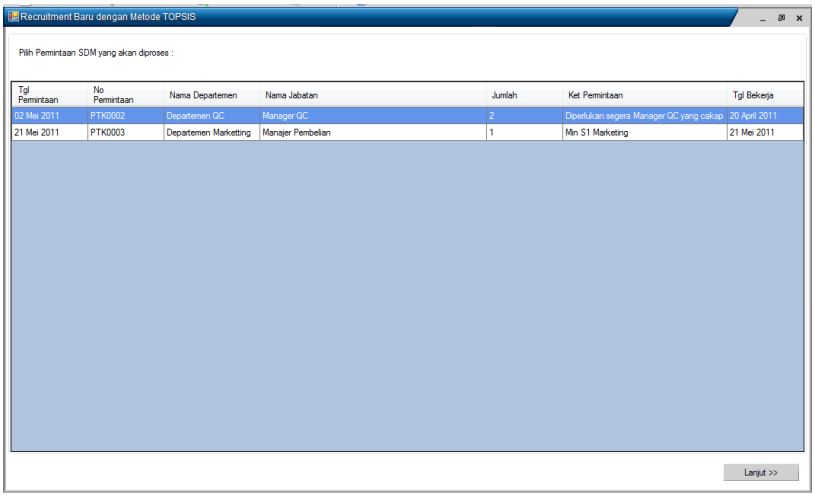
Gambar 9 Tampilan Menu Pembayaran Gaji

Pada menu recruitment TOPSIS ditampilkan data hasil recruitment pegawai dengan menggunakan DSS metode TOPSIS. User dapat memilih data permintaan SDM yang telah diproses dan melihat hasil recruitmentnya.



Gambar 10 Tampilan Menu *Recruitment* TOPSIS

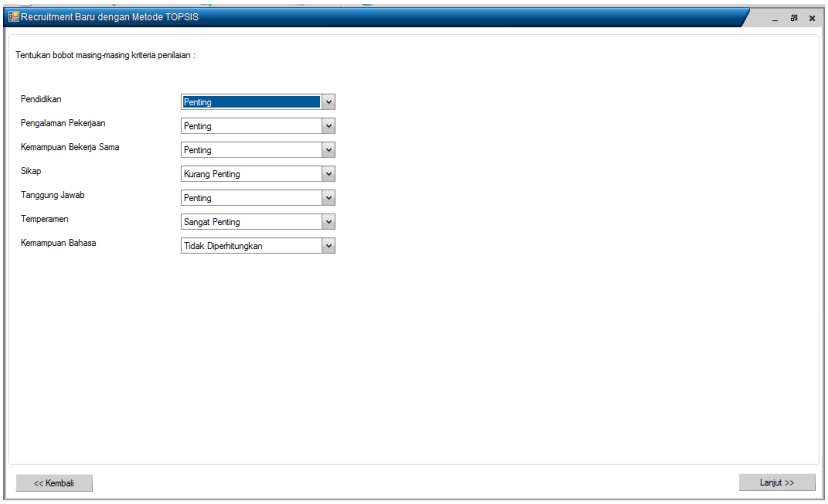
Untuk melakukan *recruitment* pegawai baru, *user* dapat menekan *button recruitment* baru. Setelah itu, akan muncul data permintaan SDM yang belum diproses. *User* dapat memilih data permintaan SDM yang akan diproses lalu menekan *button* Lanjut. Berikut ini adalah tampilan dari menu tambah *recruitment* tab permintaan SDM:



Gambar 11 Tampilan Menu Tambah *Recruitment* Tab Permintaan SDM

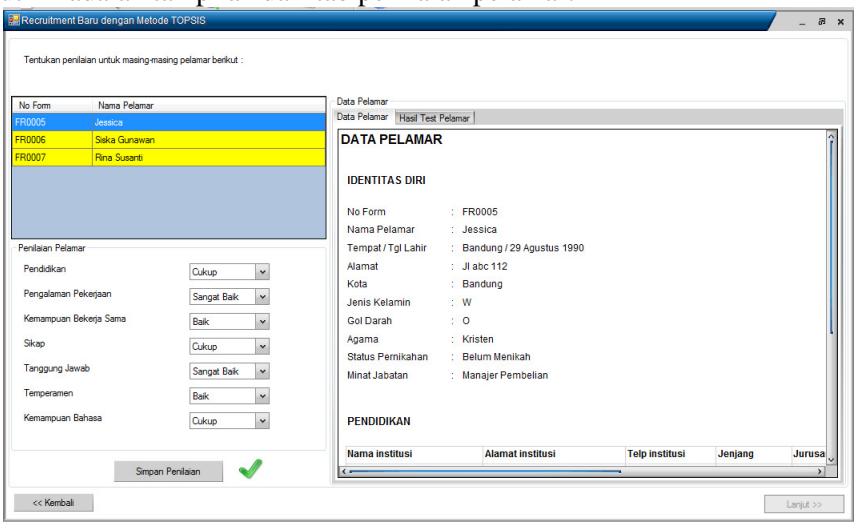
Setelah permintaan SDM dipilih, *user* dapat menentukan bobot prioritas masing-masing kriteria penilaian. Kriteria penilaian yang ditampilkan adalah *dynamic control* yang dibuat sesuai dengan data kriteria yang dimasukkan pada menu Data Kriteria. Setelah bobot prioritas ditentukan, *user* dapat menekan *button* Lanjut.

Tampilan Menu Tambah Recruitment Tab Prioritas Kriteria dapat dilihat pada gambar 14.



Gambar 12 Tampilan Menu Tambah *Recruitment* Tab Prioritas Kriteria

Setelah *user* menekan *button* Lanjut, maka *user* tampil tab penilaian pelamar. Berikut ini adalah tampilan dari tab penilaian pelamar:

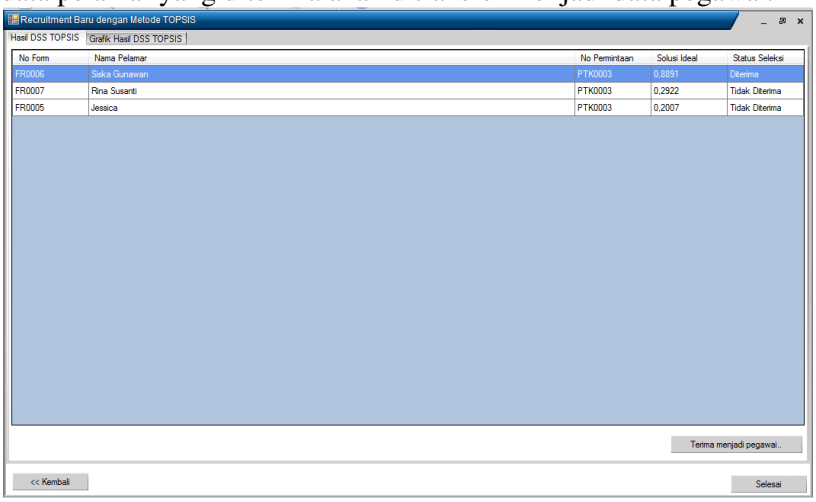


Gambar 13 Tampilan Tambah *Recruitment* Tab Penilaian Pelamar

Pada bagian penilaian pelamar, akan dimunculkan data pelamar yang melamar untuk jabatan yang diinginkan dalam permintaan SDM. *User* dapat melakukan penilaian terhadap masing-masing pelamar sesuai dengan kriteria yang ada. *User* juga dapat melihat data lengkap dan hasil test pelamar pada tab Data Pelamar dan tab Hasil Test Pelamar.

Untuk memasukkan penilaian pelamar, *user* dapat memilih pelamar yang akan dinilai, lalu menentukan hasil penilaiannya sesuai dengan kriteria yang ada. Setelah semua kriteria dinilai, *user* dapat menekan *button* Simpan penilaian. Warna kuning pada *datagrid* pelamar menunjukkan bahwa data penilaian untuk pelamar tersebut belum dimasukkan. Setelah semua data penilaian pelamar dimasukkan, *button* Lanjut dapat ditekan.

Setelah tombol Lanjut ditekan, sistem akan melakukan proses perhitungan dengan metode TOPSIS untuk menentukan nilai solusi ideal pelamar. Semakin besar nilai solusi ideal pelamar (nilai paling besar adalah satu), semakin layak pelamar tersebut diterima. Data hasil perhitungan dengan metode TOPSIS ditampilkan secara terurut dari nilai solusi ideal tertinggi. Untuk melihat grafik hasil perhitungan *user* dapat memilih tab Grafik Hasil DSS TOPSIS. Untuk menerima pelamar, *user* dapat memilih pelamar tersebut lalu memilih *button* Terima Menjadi Pegawai. Setelah semua pelamar yang diinginkan diterima, tekan *button* Selesai. Semua data pelamar yang diterima akan ditransfer menjadi data pegawai.



Gambar 14 Tampilan Menu Tambah *Recruitment* Tab Hasil DSS

VI. Kesimpulan dan Saran

Aplikasi sistem informasi manajemen human resources ini memiliki kemampuan sebagai berikut :

1. Melakukan proses *recruiment* pegawai secara terintegrasi mulai dari permintaan SDM sampai kepada penerimaan pegawai.
2. Melakukan pengolahan data pegawai dan pelamar secara mudah dan terintegrasi.
3. *Import* data pelamar dari file excel.
4. *Import* data pelamar yang diterima menjadi data pegawai.
5. Melakukan proses absensi dengan password dan juga foto absensi dengan *webcam* untuk menghindari kecurangan.

6. Memproses data cuti,ijin, hari libur dan pelanggaran pegawai.
7. Melakukan perhitungan gaji, THR serta pesangon secara otomatis.

Selain memiliki kemampuan tersebut aplikasi sistem informasi ini juga memiliki beberapa kekurangan yang harus diperhatikan, yaitu :

1. Perlu adanya penambahan fitur agar aplikasi dapat diakses di banyak komputer atau secara *online*.
2. Perlu adanya penambahan fitur absensi dengan menggunakan sidik jari agar lebih meningkatkan keamanan dan akurasi data.
3. Perlu adanya penambahan fitur agar aplikasi dapat diakses melalui handphone atau alat *mobile* lainnya, dengan membuat aplikasi mobile yang dapat mengakses aplikasi.

Daftar Pustaka

Ayuliana. (2009). *Black box*. Retrieved June 20, 2010, from *http://ayuliana_st.staff.gunadarma.ac.id/.../Pertemuan+06+-++_BlackBox+Testing_.pdf*

Chonoles, Michael Jesse dan Schardt, James A.(2003). *UML 2 for Dummies*. Hoboken:Wiley Publishing

Febriani. (2007). *Flowchart*. Retrieved June 20, 2010, from *febriani.staff.gunadarma.ac.id/Downloads/files/5616/Flowchart.pdf*

Hasibuan, Melayu S.P. (2006). *Manajemen Sumber Daya Manusia*. Jakarta: PT Bumi Aksara

Hwang, Ching-Lai dan Kwangsun Yoon. (1981). *Multiple Attribute Decision Making, Methods and Application.* Berlin:Springer-Verlag

Kadarsah Suryadi, Ramdhani Ali. (1998). *Sistem Pendukung Keputusan*. Bandung : Remaja Rosdakarya

Noviyanto. (2009) . *Sistem Informasi Sumber Daya Manusia (SISDM/HRIS)*. Retrieved September 22, 2010, from *http://viyan.staff.gunadarma.ac.id/Downloads/files/16480/10_Sistem+InformaIn+Sumber+Daya+Manusia.pdf*

Pipiew.(2007). *Proses Bisnis*. Retrieved June 20, 2010,from *http://pipiew.wordpress.com/2007/11/29/proses-bisnis/*

Primashanti, Ida Ayu Y.(2007). *Entity Relationship Diagram*. Retrieved June 20, 2010 ,from *http://iaprima.staff.gunadarma.ac.id/Downloadsfiles/5459/Bahasan9a_ERD.pdf*

Sutabri, Tata. (2003). *Analisa Sistem Informasi*, *Edisi Pertama.* Yogyakarta: Andi.