

SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN KARYAWAN TERBAIK PADA LAMOIST LAYERS BATAM

Indra Gunawan^{*1}, Afrina², Cici Sofrawida³

^{1,2,3}STT Ibnu Sina; Jl. Teuku Umar - Lubuk Baja; telp/fax: 0778-425391/0778-458394

^{1,2,3}Program Studi Teknik Informatika, STT Ibnu Sina, Batam

e-mail: ^{*1}indra.gunawan@stt-ibnusina.ac.id, ²afrina@stt-ibnusina.ac.id,

³1410128262052@stt-ibnusina.ac.id

Abstrak

Lamoist Layers Batam merupakan salah satu Perusahaan yang bergerak di bidang produksi dan pemasaran oleh-oleh kue lapis khususnya di Kota Batam dan sudah berkiprah didunia industri selama 10 tahun dan sangat terkenal dinegara tetangga seperti Singapore dan Malaysia. Dalam melaksanakan operasional Perusahaan Lamoist Layers Batam memberikan penghargaan kepada karyawan dengan cara memilih karyawan terbaik setiap bulan. Tujuan dari penelitian ini adalah untuk merancang aplikasi sistem pendukung keputusan pemilihan karyawan terbaik pada lamoist layers batam dengan menerapkan Metode Simple Additive Weighting dengan kriteria C1: Disiplin, C2: Kerja Tim, C3: Hasil Opname, C4: Target Penjualan, C5: Good Service dan C6: Laporan Penjualan sehingga didapatkan V1 sebagai hasil akhir penilaian karyawan terbaik dengan nilai 100. Objek dalam penelitian ini adalah pemilihan karyawan terbaik pada lamoist layers batam menggunakan data karyawan pada bulan Januari sampai Maret 2018. Rancangan sistem ini menggunakan metode Object Oriented analysis design dengan metode pemodelan Unified Modeling Language serta implementasi sistem menggunakan sistem database MySQL dan bahasa pemrograman PHP. Hasil dari penelitian ini diharapkan mampu diterapkan pada perusahaan dalam melakukan proses pemilihan karyawan terbaik setiap bulan.

Kata kunci—Sistem Pendukung Keputusan, Simple additive Weighting, Object Oriented analysis design

Abstract

Lamoist layers batam is one of the companies engaged in the production and marketing of layers cake specially in Batam City and has been active in the industrial world for 10 years and is very well known in neighboring countries such as Singapore and Malaysia. In operations company lamoist layers batam give rewards for employees with choosing the best employees every month. This research designs a decision support system for selecting the best employees in lamoist layers batam. The purpose of this study was to design a decision support system application for the selection of the best employees in lamoist layers batam apply Simple Additive Weighting Method with criteria C1: Discipline, C2: Team Work, C3: Results Opname, C4: Sales Target, C5: Good Service and C6: Sales Report so that V1 is obtained as the final result of the best employee evaluation with a value of 100. The object in this study is the selection of the best employees in the lamoist layers batam using employee data in January - March 2018. This system design uses Object Oriented analysis design method with Unified Modeling Language modeling method and system implementation using My-SQL database system and PHP programming language. The results of this research are expected to be able to be applied to companies in the process of selecting the best employees every month.

Keywords—Decision Support System, Simple Additive Weighting, Object Oriented analysis design

1. PENDAHULUAN

Lamoist layers batam merupakan salah satu perusahaan yang bergerak di bidang produksi dan pemasaran oleh-oleh kue lapis khususnya di kota batam dan sudah berkiprah di dunia industri selama 10 tahun dan sangat terkenal dinegara tetangga seperti Singapore dan Malaysia.

Dalam melaksanakan operasional perusahaan Lamoist Layers Batam memberikan penghargaan kepada karyawan dengan cara memilih karyawan terbaik di setiap bulannya, hal ini dimaksudkan untuk meningkatkan semangat karyawan dalam bekerja, terutama dalam memberikan pelayanan terbaik kepada pelanggan, proses pemilihan karyawan terbaik pada Lamoist layers batam saat ini masih sangat lambat dan memiliki hasil yang kurang tepat, ketidak tepatan dalam memberikan nilai kepada karyawan berdampak pada hasil keputusan yang kurang memuaskan bagi karyawan itu sendiri.

Permasalahan di atas dapat diperbaiki dengan membangun suatu Sistem Pendukung Keputusan Penilaian Karyawan Terbaik Pada Lamoist layers batam. Penelitian ini menggunakan metode *Simple Additive weighting* (SAW) yang sering dikenal dengan metode penjumlahan berbobot, karena mampu menyeleksi alternatif terbaik dari sejumlah alternatif berdasarkan kriteria-kriteria.

I Gede Bendesa Subawa, I Made Agus Wirawan, I Made Gede Sunarya (2015) metode *Simple Additive Weighting* digunakan untuk pengembangan sistem pendukung keputusan pemilihan pegawai terbaik di PT. Tirta Jaya Abadi Singaraja yang telah lulus uji, mendapatkan hasil yang baik dan dapat diterapkan. Wahyu adam, M.Eng.Sc dan Asep Rohaendi (2017), sistem pendukung keputusan pemilihan karyawan terbaik menggunakan Metode *Simple Additive Weighting* di PT. Eltran Indonesia berhasil membantu perusahaan dalam memberikan penghargaan kepada pegawai serta bisa melakukan evaluasi untuk memajukan perusahaan menjadi lebih baik.

Berdasarkan referensi di atas maka penulis tertarik menggunakan Metode *Simple Additive Weighting* (SAW) untuk menentukan karyawan terbaik pada Lamoist Layers Batam.

2. METODE PENELITIAN

2.1 Sistem Pendukung Keputusan

(Fitri & Nurhadi, 2017) Konsep Sistem Pendukung Keputusan (SPK) atau *Decision Support Systems* (DSS) pertama kali diperkenalkan oleh Michael S. Scott Morton pada awal tahun 1970-an, yang selanjutnya dikenal dengan istilah *Management Decision Systems*. Pada proses pengambilan keputusan, pengolahan data dan informasi yang dilakukan bertujuan untuk menghasilkan berbagai alternatif keputusan yang dapat diambil.

SPK yang merupakan penerapan dari sistem informasi ditujukan hanya sebagai alat bantu manajemen dalam pengambilan keputusan. SPK tidak dimaksudkan untuk menggantikan fungsi pengambil keputusan dalam membuat keputusan, melainkan hanyalah sebagai alat bantu pengambil keputusan dalam melaksanakan tugasnya. SPK dirancang untuk menghasilkan berbagai alternatif yang ditawarkan kepada para pengambil keputusan dalam melaksanakan tugasnya. Sehingga dapat dikatakan bahwa SPK memberikan manfaat bagi manajemen dalam hal meningkatkan efektivitas dan efisiensi kerjanya terutama dalam proses pengambilan keputusan. Di samping itu, SPK menyatukan kemampuan komputer dalam pelayanan interaktif terhadap penggunaanya dengan adanya proses pengolahan atau manipulasi data yang memanfaatkan model atau aturan yang tidak terstruktur sehingga menghasilkan alternatif keputusan yang situasional.

Fitri dan Nurhadi (2017) Sistem pendukung keputusan dapat memberikan berbagai manfaat atau keuntungan bagi pemakainya, antara lain:

1. Memperluas kemampuan pengambilan keputusan dalam memproses data/informasi bagi pemakainya.

2. Membantu pengambilan keputusan dalam hal penghematan waktu yang dibutuhkan untuk memecahkan masalah terutama berbagai masalah yang sangat kompleks dan tidak terstruktur.
3. Dapat menghasilkan solusi dengan lebih cepat serta hasilnya dapat diandalkan.
4. Walaupun suatu Sistem Pendukung Keputusan, mungkin saja tidak mampu memecahkan masalah yang dihadapi oleh pengambil keputusan, namun dapat menjadi stimulan bagi pengambil keputusan dalam memahami persoalannya, karena sistem pendukung keputusan mampu menyajikan berbagai alternatif.

2.2 *Metode Pengumpulan Data*

Walsham (1995) yang dikutip oleh (Gunawan, 2016) menjelaskan bahwa metode pengumpulan data penelitian kualitatif menggunakan metode wawancara secara mendalam.

Tahap-tahap pengumpulan data dalam penelitian ini adalah sebagai berikut:

- a. Studi Pustaka
Metode studi pustaka adalah suatu teknik atau cara pengumpulan data atau informasi yang dilakukan dengan cara membaca buku-buku, laporan-laporan yang berhubungan dengan masalah terkait.
- b. Wawancara
Wawancara adalah metode pengumpulan data yang dilakukan dengan cara tanya jawab secara langsung Sales Coordinator PT.Lamoist Layers Batam.
- c. Observasi
Metode observasi adalah salah satu teknik pengumpulan data atau informasi yang dilakukan dengan cara melakukan pengamatan langsung di PT.Lamoist Layers Batam.
- d. Dokumentasi
Metode dokumentasi adalah metode pengumpulan data yang berupa catatan, transkrip, buku, surat kabar, majalah, notulen rapat, legger, agenda dan sebagainya yang ada di PT Lamoist Layers Batam.

2.3 *Metode Analisa Data*

Data yang telah diolah kemudian dianalisis dengan menggunakan analisis deskriptif kualitatif. Analisis deskriptif kualitatif adalah analisis yang diwujudkan dengan cara menggambarkan kenyataan atau keadaan-keadaan atas suatu obyek dalam bentuk uraian kalimat berdasarkan keterangan-keterangan dari pihak-pihak yang berhubungan langsung dengan penelitian ini. Hasil analisis tersebut kemudian diinterpretasikan guna memberikan gambaran yang jelas terhadap permasalahan yang diajukan mengenai Sistem Pendukung Keputusan Pemilihan Karyawan Terbaik (Studi Kasus Lamoist Layers Batam).

2.4 *Metode Pengolahan Data*

Ladjamudin (2013), Pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi formasi yang memiliki kegunaan. Adapun metode pengolahan data yang penulis gunakan dalam penelitian kali ini yaitu metode kualitatif dengan menggunakan data Kuantitatif (angka) data kualitatif (bukan angka), Sugiyono (2010:15), menjelaskan bahwa Metode penelitian kualitatif merupakan metode penelitian yang berlandaskan pada filsafat positivisme, digunakan untuk meneliti pada kondisi objek yang alamiah, (sebagai lawannya adalah eksperimen) dimana peneliti adalah sebagai instrumen kunci, pengambilan sampel sumber data dilakukan secara *purposive* dan snowbaal, teknik pengumpulan dengan triangulasi, analisis data bersifat induktif, kualitatif/kuantitatif dan hasil penelitian kualitatif lebih menekankan makna dari pada generalisasi. pengolahan data yang dilakukan pada penelitian ini adalah meliputi:

1. studi pustaka.
2. mengumpulkan data.
3. menganalisa data.
4. mengidentifikasi masalah dan kebutuhan.

5. melakukan desain.
6. Implementasi.
7. pengujian sistem.
8. menyajikan dalam bentuk website.

2.5 Metode Pemecahan Masalah

Metode pemecahan masalah dalam penelitian ini menggunakan metode *simple additive weighting* dengan tahapan sebagai berikut:

1. Menentukan kriteria-kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu C_i .
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan kriteria(C_i).
4. Melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan ataupun atribut biaya) sehingga diperoleh matriks ternormalisasi R.

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\max x_{ij}} & \text{Jika } j \text{ adalah atribut Keuntungan (1)} \\ \frac{\min x_{ij}}{x_{ij}} & \text{Jika } j \text{ adalah atribut Cost (2)} \end{cases}$$

Hasil akhir diperoleh dari proses perankingan yaitu penjumlahan dari perkalian matriks ternormalisasi R dengan vektor bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik sebagai solusi dengan rumus :

$$V_i = \sum_{j=1}^n w_j R_{ij} \quad (3)$$

Keterangan :

V_i : Rangkaian untuk setiap alternatif

W_j : Nilai bobot dari setiap kriteria

R_{ij} : Nilai rating kinerja ternormalisasi Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih.

2.6 Metode Perancangan Sistem

Menurut Booch's yang dikutip dalam Prasetyotomo (2015) Metodologi Booch's *Object Oriented Analysis & Design*, selanjutnya disingkat OOAD, merupakan penggabungan 3 (tiga) pendekatan, yaitu: *Object Oriented Design* (OOD), *Object Oriented Analysis* (OOA) dan *Object Oriented Programming* (OOP).

OOA merupakan metode analisis yang memeriksa kebutuhan (requirement) berdasarkan perspektif pengumpulan obyek-obyek dan kelas-kelas dalam sebuah domain problem, OOD merupakan sebuah metode mendesain yang mencakup proses pendekomposisi obyek dan digambarkan dalam notasi sehingga bisa menggambarkan static dan dynamic model sistem baik secara logical dan/atau physical, sedangkan OOP merupakan sebuah metode untuk mengimplementasikan program yang diorganisasikan sebagai kumpulan obyek dimana tiap-tiap obyek merupakan instan dari sebuah kelas dan kelas merupakan salah satu dari kumpulan kelas yang saling berhubungan secara hirarki menggunakan *inheritance relationship*.

Hubungan antara OOA, OOD dan OOP adalah: hasil pemodelan atau pengumpulan obyek dari OOA akan digunakan oleh OOD dan hasil dari OOD akan digunakan sebagai blueprint untuk membangun sistem dengan menggunakan OOP.

Tahapan dalam proses OOAD adalah sebagai berikut :

a. Requirement

Pada tahap *requirement* menjelaskan tentang bagaimana membuat dan menjaga sebuah perjanjian kerjasama dengan *customer* dan stakeholder mengenai apa yang harus dilakukan pada pembuatan sistem, misalnya kebutuhan-kebutuhan dari suatu sistem seperti fungsi-fungsi apa saja yang akan digunakan oleh sistem, fitur-fitur apa saja yang akan dipakai dalam pembuatan sistem, siapa saja user yang terlibat atau yang memakai sistem tersebut dan lain sebagainya.

b. *Analysis and Design*

Tahapan ini menjelaskan bagaimana mengkonversi kebutuhan- kebutuhan sistem yang telah dibuat sebelumnya menjadi sebuah bentuk rancangan sistem. Yang mana rancangan tersebut disajikan sebagai spesifikasi dari implementasi sistem dalam memilih lingkungan penerapannya. Selain itu, tahapan analisis dan desain juga mencakup dalam mengembangkan sebuah arsitektur/bentuk rancangan yang kokoh.

c. *Implementation*

Setelah membuat suatu kebutuhan sistem, menganalisis dan merancang sistem yang akan dibuat. Kemudian pada tahap ini dilakukan implementasi unit sistem dan menggabungkan rancangan dengan sistem, maksudnya adalah mengimplementasikan rancangan tersebut dalam sebuah sistem atau program dalam bentuk *coding* program. Selain itu juga menghasilkan sebuah sistem yang sudah dapat dijalankan.

d. *Testing*

Tes/pengujian dilakukan untuk meyakinkan bahwa sistem telah sesuai dengan kebutuhan-kebutuhan yang telah dibuat sebelumnya (kebutuhan-kebutuhan tersebut adalah menerapkan yang sewajarnya). Memvalidasi fungsi sistem yang telah konkrit kemudian didemonstrasikan bahwa produk *software* tersebut sesuai dengan kebutuhan dan rancangan yang telah dibuat sebelumnya.

e. *Deployment*

Meyakinkan bahwa produk *software* tersebut (termasuk implementasi dan pengujian) telah tersedia untuk *end user* atau sudah dapat digunakan oleh pengguna akhir.

2.7 Metode Pemodelan Sistem

Unified Modelling Language (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual (Braun,et.al,2001) yang dikutip dalam Ropianto (2016). Juga merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek (Whitten,et.al,2004) yang dikutip dalam Ropianto (2016).

Jenis-jenis Diagram Pada UML

a. *Use Case Diagram*

Use Case diagram merupakan uraian sekelompok yang saling terkait dan membentuk system secara teratur yang dilakukan atau diawasi oleh sebuah actor. Use Case atau diagram use case merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat (Rossa M.Salahuddin, 2011) yang dikutip dalam Nuddin & Fithri (2015).


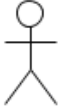



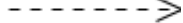

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya *login* ke sistem, membuat sebuah daftar belanja, dan sebagainya. Sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

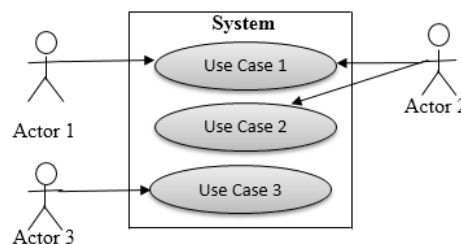
Use case diagram dapat sangat membantu bila kita sedang menyusun requirement sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua fitur yang ada pada sistem.

Sebuah *use case* dapat meng-include fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang di-include akan dipanggil setiap kali *use case* yang meng-include dieksekusi secara normal. Sebuah *use case* dapat di-include oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*.

Sebuah *use case* juga dapat meng-extend *use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.

Tabel 1 Simbol Use case Diagram (A.S & Shalahuddin 2011:130) yang dikutip oleh Aprianti & Maliha (2017)

GAMBAR	DESKRIPSI
<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal-awal frase nama <i>use case</i> .
Aktor/ <i>Actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, tapi actor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.
Asosiasi/ <i>Association</i> 	Komunikasi actor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor.
Ekstensi/ <i>Extend</i> <<Extend>> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip inheritance pada pemrograman berorientasi objek.
Generalisasi/ <i>Generalization</i> 	Hubungan generalisasi dan spesialisasi (umum- khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lain.
Menggunakan/ <i>Include/Uses</i> <<include>>  <<uses>> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.





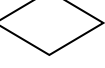


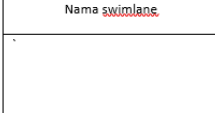
Gambar 1 Contoh Use Case Diagram (Ropianto, 2016)

b. Activity diagram

Activity Diagram menggambarkan workflow (aliran kerja) dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. (Aprianti & Maliha,2017) *Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. Sama seperti *state*, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik inkronisasi yang dapat berupa titik, garis horizontal atau vertikal. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

Tabel 2 Simbol *Activity diagram* (A.S & Shalahuddin 2011:134)
yang dikutip oleh Aprianti & Maliha (2017)

GAMBAR	DESKRIPSI
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktifitas

c. Class Diagram

Diagram kelas atau Class diagram menggambarkan sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. (Rossa M.Salahuddin, 2011) pada jurnal (Nuddin & Fithri, 2015).

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class* memiliki tiga area pokok:

1. Nama (dan *stereotype*)
2. Atribut
3. Metode

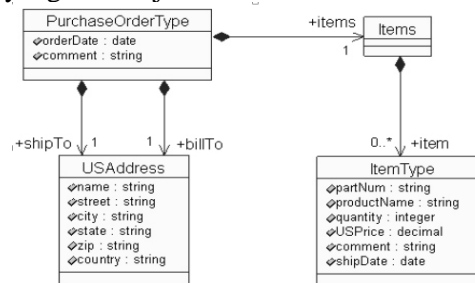
Atribut dan metoda dapat memiliki salah satu sifat berikut:

1. *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan.
2. *Protected*, hanya dapat dipanggil oleh *class* yang bersangkutan dan anak-anak yang mewarisinya.
3. *Public*, dapat dipanggil oleh siapa saja.

Class dapat merupakan implementasi dari sebuah *interface*, yaitu *class* abstrak yang hanya memiliki metoda. *Interface* tidak dapat langsung diinstansiasikan, tetapi harus diimplementasikan dahulu menjadi sebuah *class*. Dengan demikian *interface* mendukung resolusi metoda pada saat *run-time*.

Berikut ini merupakan hubungan antar *class*:

1. Asosiasi, yaitu hubungan statis antar *class*. Umumnya menggambarkan *class* yang memiliki atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain. Panah *navigability* menunjukkan arah *query* antar *class*.
2. Agregasi, yaitu hubungan yang menyatakan bagian (“terdiri atas”).
3. Pewarisan, yaitu hubungan hirarki antar *class*. *Class* dapat diturunkan dari *class* lain dan mewarisi semua atribut dan metoda *class* asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari *class* yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
4. Hubungan dinamis, yaitu rangkaian pesan (*message*) yang di-*passing* dari satu *class* kepada *class* lain. Hubungan dinamis dapat digambarkan dengan menggunakan *sequence diagram* yang akan dijelaskan kemudian.

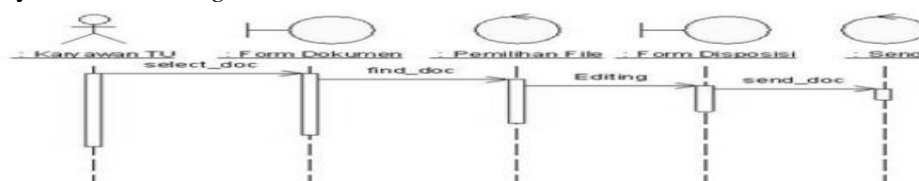


Gambar 2. Contoh *Class Diagram* (Ropianto, 2016)

d. *Sequence Diagram*

Sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence diagram* adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan use case diagram (Ropianto, 2016). *Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah *message*.



Gambar 2.3 Notasi *Sequence Diagram* (Ropianto, 2016)

e. *User Interface*

merupakan komponen penting dari setiap sistem baru yang akan digunakan. *User Interface* merupakan bagian dari sistem informasi yang memerlukan interaksi langsung pada sistem *user*

dalam membuat *input* dan *output*. Dalam membuat user *interface* terdapat aturan yang disebut *eight golden rules*. *Eight golden rules* bermanfaat dalam merancang *user interface* yang interaktif.

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Data

Berdasarkan Observasi secara langsung di Lamoist Layers Batam dan wawancara dengan Sales Coordinator Lamoist Layers Batam, maka diperoleh data yang akan diolah untuk perancangan sistem pendukung keputusan pemilihan karyawan terbaik pada Lamoist Layers Batam.

Data primer yang di dapatkan dari hasil wawancara secara mendalam sebagai berikut:

1. Data kriteria yang akan digunakan pada penilaian karyawan seperti Disiplin, hasil opname sesuai, Ringkasan Penjualan Harian, *Good Service* dari *Customer*, Kerja Sama Tim, Target Penjualan.
2. Data nilai dari bobot dari kriteria-kriteria yang akan digunakan dalam penilaian karyawan terbaik.

Data sekunder yang didapatkan dari hasil observasi secara langsung sebagai berikut:

1. Data Karyawan Lamoist Layers Batam.
2. Data Ringkasan Penjualan Harian Lamoist Layers Batam.

3.2 Tahapan pemecahan masalah menggunakan metode *Simple Additive Weighting*.

Berikut adalah tahapan pemecahan masalah menggunakan metode *Simple Additive Weighting* adalah sebagai berikut:

1. Penentuan Kriteria dan Bobot

Nilai bobot pada setiap kriteria serta himpunan kriteria ditentukan oleh manajemen perusahaan sebagai pemegang keputusan. Bobot dan kriteria yang ditentukan dapat dilihat pada tabel 3.

Tabel 3 Tabel Nama Kriteria dan Bobot

Kriteria	Keterangan	Bobot
C1	Disiplin (C1)	20
C2	Kerja Tim (C2)	10
C3	Hasil Opname (C3)	10
C4	Target Penjualan (C4)	30
C5	<i>Good Service</i> (C5)	10
C6	Laporan Penjualan (C6)	20

C = Kriteria

2. Menentukan rating kecocokan setiap alternatif pada kriteria. Rating kecocokan setiap kriteria diperoleh dari hasil observasi pihak manajemen yang diwakili *sales coordinator* terhadap seluruh karyawan yang akan dijadikan kandidat kriteria.

Tabel 4 Tabel Rating Kecocokan Setiap Alternatif pada Kriteria

kepentingan	20	10	10	30	10	20
alternatif / kriteria	Disiplin	Kerja Tim	Hasil Opname	Target Hasil	Good Service	Laporan penjualan
Siti	80	100	50	50	60	50
Rahma	60	60	20	50	60	50
Elsa	100	100	100	100	100	100
Caca	80	60	100	50	80	100
Sri	60	80	50	50	40	50
Pembagi	60	60	100	50	40	50

3. Membuat Matriks keputusan berdasarkan rating kecocokan setiap alternatif pada kriteria.

$$\text{Matriks } X = \begin{pmatrix} 80 & 100 & 50 & 50 & 60 & 50 \\ 60 & 60 & 20 & 50 & 60 & 50 \\ 100 & 100 & 100 & 100 & 100 & 100 \\ 80 & 60 & 100 & 50 & 80 & 100 \\ 60 & 80 & 50 & 50 & 40 & 50 \end{pmatrix}$$

4. Melakukan Normalisasi matriks ternormalisasi R.

- a. Normalisasi matriks baris 1-5 pada kolom 1

$$r_{11} = \frac{80}{\text{Max } 80;60;100;80;60} = \frac{80}{100} = 0.8$$

$$r_{21} = \frac{60}{\text{Max } 80;60;100;80;60} = \frac{60}{100} = 0.6$$

$$r_{31} = \frac{100}{\text{Max } 80;60;100;80;60} = \frac{100}{100} = 1$$

$$r_{41} = \frac{80}{\text{Max } 80;60;100;80;60} = \frac{80}{100} = 0.8$$

$$r_{51} = \frac{60}{\text{Max } 80;60;100;80;60} = \frac{60}{100} = 0.6$$

- b. Normalisasi matriks baris 1-5 pada kolom 2

$$r_{12} = \frac{100}{\text{Max } 100;60;100;60;80} = \frac{100}{100} = 1$$

$$r_{22} = \frac{60}{\text{Max } 100;60;100;60;80} = \frac{60}{100} = 0.6$$

$$r_{32} = \frac{100}{\text{Max } 100;60;100;60;80} = \frac{100}{100} = 1$$

$$r_{42} = \frac{60}{\text{Max } 100;60;100;60;80} = \frac{60}{100} = 0.6$$

$$r_{52} = \frac{80}{\text{Max } 100;60;100;60;80} = \frac{80}{100} = 0.8$$

- c. Normalisasi matriks baris 1-5 pada kolom 3

$$r_{13} = \frac{50}{\text{Max } 50;20;100;100;50} = \frac{50}{100} = 0.5$$

$$r_{23} = \frac{20}{\text{Max } 50;20;100;100;50} = \frac{20}{100} = 0.2$$

$$r_{33} = \frac{100}{\text{Max } 50;20;100;100;50} = \frac{100}{100} = 1$$

$$r_{43} = \frac{100}{\text{Max } 50;20;100;100;50} = \frac{100}{100} = 1$$

$$r_{53} = \frac{50}{\text{Max } 50;20;100;100;50} = \frac{50}{100} = 0.5$$

- d. Normalisasi matriks baris 1-5 pada kolom 4

$$r_{14} = \frac{50}{\text{Max } 50;50;100;50;50} = \frac{50}{100} = 0.5$$

$$r_{24} = \frac{50}{\text{Max } 50;50;100;50;50} = \frac{50}{100} = 0.5$$

$$r_{34} = \frac{100}{\text{Max } 50;50;100;50;50} = \frac{100}{100} = 1$$

$$r_{44} = \frac{50}{\text{Max } 50;50;100;50;50} = \frac{50}{100} = 0.5$$

$$r_{54} = \frac{50}{\text{Max } 50;50;100;50;50} = \frac{50}{100} = 0.5$$

- e. Normalisasi matriks baris 1-5 pada kolom 5

$$r_{15} = \frac{60}{\text{Max } 60;60;100;80;40} = \frac{60}{100} = 0.6$$

$$r_{25} = \frac{60}{\text{Max } 60;60;100;80;40} = \frac{60}{100} = 0.6$$

$$r_{35} = \frac{100}{\text{Max } 60;60;100;80;40} = \frac{100}{100} = 1$$

$$r_{45} = \frac{80}{\text{Max } 60;60;100;80;40} = \frac{80}{100} = 0.8$$

$$r_{55} = \frac{40}{\text{Max } 60;60;100;80;40} = \frac{40}{100} = 0.4$$

- f. Normalisasi matriks baris 1-5 pada kolom 6

$$r_{16} = \frac{50}{\text{Max } 50;50;100;100;50} = \frac{50}{100} = 0.5$$

$$r_{26} = \frac{50}{\text{Max } 50;50;100;100;50} = \frac{50}{100} = 0.5$$

$$r_{36} = \frac{100}{\text{Max } 50;50;100;100;50} = \frac{100}{100} = 1$$

$$r_{46} = \frac{100}{\text{Max } 50;50;100;100;50} = \frac{100}{100} = 1$$

$$r_{56} = \frac{50}{\text{Max } 50;50;100;100;50} = \frac{50}{100} = 0.5$$

Tabel 5 Tabel Matriks Ternormalisasi R

Alternatif	Normalisasi					
Siti	0,8	1	0,5	0,5	0,6	0,5
Rahma	0,6	0,6	0,2	0,5	0,6	0,5
Elsa	1	1	1	1	1	1
Caca	0,8	0,6	1	0,5	0,8	1
Sri	0,6	0,8	0,5	0,5	0,4	0,5

5. perkalian matriks ternormalisasi R dengan vektor bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik (A_i) sebagai solusi.
Vektor bobot Kriteria

$$\begin{aligned} C1 &= 20 & C3 &= 10C5 = 10C4 = 30 \\ C5 &= 10 & C6 &= 20 \end{aligned}$$

Tabel 6 Tabel Perkalian Matriks Ternormalisasi R dengan Vektor Bobot

Alternatif	C1	C2	C3	C4	C5	C6
Siti	$20 \times 0,8$	10×1	$10 \times 0,5$	$30 \times 0,6$	$10 \times 0,5$	$20 \times 0,5$
Rahma	$20 \times 0,6$	$10 \times 0,6$	$10 \times 0,2$	$30 \times 0,5$	$10 \times 0,6$	$20 \times 0,5$
Elsa	20×1	10×1	10×1	30×1	10×1	20×1
Caca	$20 \times 0,8$	$10 \times 0,6$	10×1	$30 \times 0,5$	$10 \times 0,8$	20×1
Sri	$20 \times 0,6$	$10 \times 0,8$	$10 \times 0,5$	$30 \times 0,5$	$10 \times 0,4$	$20 \times 0,5$

Tabel 7. Tabel Hasil Perkalian Matriks Ternormalisasi R dengan Vektor Bobot

Alternatif	C1	C2	C3	C4	C5	C6	Total	Rangking
Siti	16	10	5	15	6	10	16	62
Rahma	12	6	2	15	6	10	12	51
Elsa	20	10	10	30	10	20	20	100
Caca	16	6	10	15	8	20	16	75
Sri	12	8	5	15	4	10	12	54

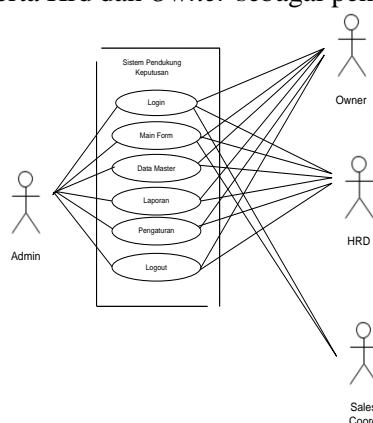
Jadi dari Perangkingan diperoleh alternatif Elsa sebagai V1 dengan nilai 100 sebagai karyawan terbaik untuk selanjutnya akan di implementasi ke program sistem pendukung keputusan pemilihan karyawan terbaik pada Lamoist Layers Batam.

3.3 Pengembangan Sistem

Metode yang digunakan dalam pengembangan sistem adalah metode *Object Oriented Analysis and Design* (OOAD) dengan metode pemodelan UML (*Unified Modeling Language*).

3.3.1 Requirement

Penerapan sistem pendukung keputusan yang diusulkan di Lamoist Layers Batam bertujuan untuk memecahkan permasalahan proses pemilihan karyawan terbaik lamoist layers batam yang masih lambat dan memiliki hasil yang kurang tepat, Sistem pendukung keputusan ini merupakan sebuah sistem yang dapat membantu manajemen dalam melakukan pemilihan karyawan terbaik dengan cara memberikan alternatif karyawan yang akan dijadikan kandidat. Sistem pendukung keputusan ini dirancang dengan menyesuaikan proses bisnis dan infrastruktur pada lamoist layers batam, tahap *requerment* ini menjelaskan bagian manajemen/operasional yang akan terlibat langsung dengan sistem ini adalah Sales Coordinator yang akan melakukan penilaian secara manual terhadap karyawan pemasaran, Admin yang akan bertugas melakukan pengimputan kedalam sistem, serta Hrd dan Owner sebagai pengawas proses berjalannya sistem.

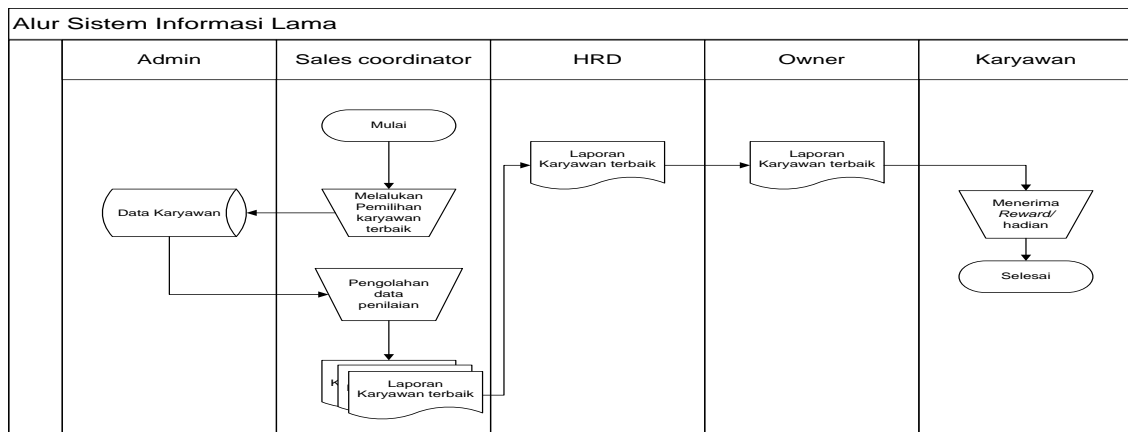


Gambar 3 Interaksi Fungsi-fungsi Sistem dan Aktor

3.3.2 Analysis dan Design

a. Analisa sistem berjalan

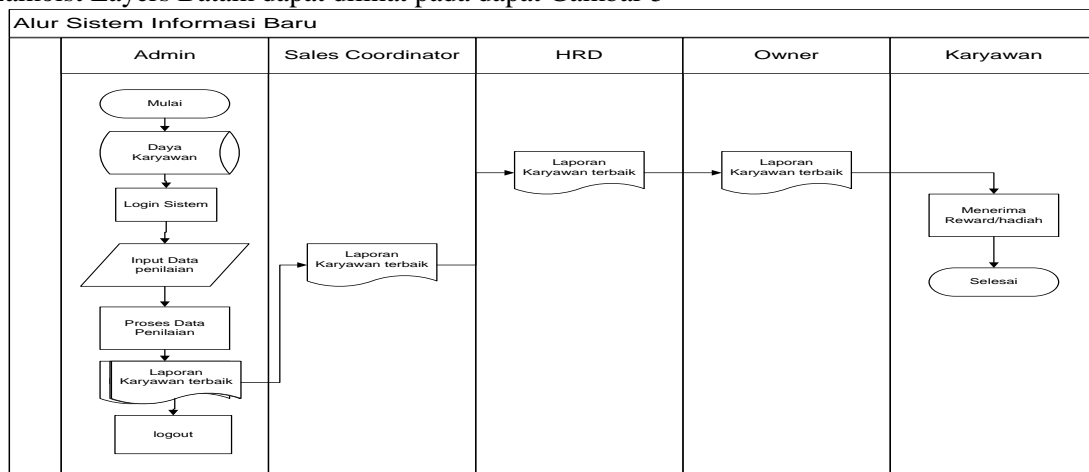
Gambaran Analisa sistem yang berjalan pemilihan karyawan terbaik pada Lamoist Layers Batam dilihat pada dapat Gambar 4



Gambar 4 Analisa sistem berjalan

b. Analisa Sistem yang Diusulkan

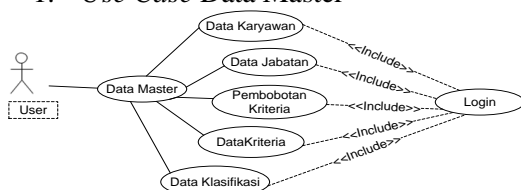
Gambaran Analisa sistem yang diusulkan pada sistem pemilihan karyawan terbaik Lamoist Layers Batam dapat dilihat pada dapat Gambar 5



Gambar 5 Analisa Sistem yang Diusulkan

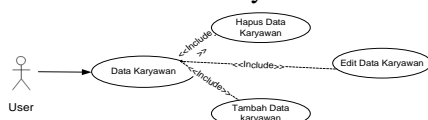
c. Ruang Lingkup Sistem yang diusulkan

1. Use Case Data Master



Gambar 6 Use Case Data Master.

2. Use Case Data Karyawan



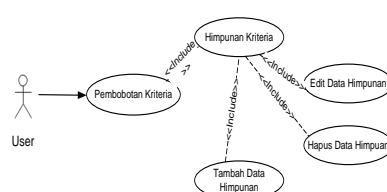
Gambar 7 Use Case Data Karyawan

3. Use Case Data Kriteria

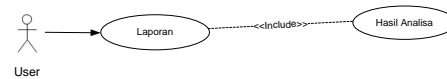


Gambar 8 Use Case Data Kriteria

4. Use Case Data Pembobotan Kriteria



. Gambar 9 Use Case Pembobotan Kriteria



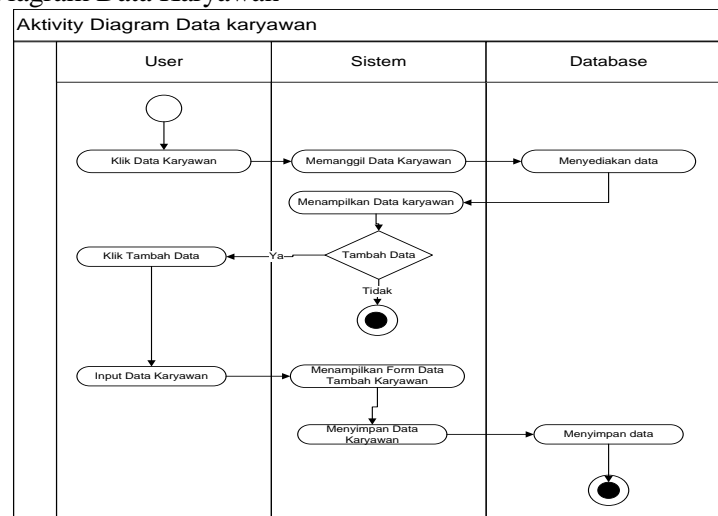
5. Use Case Laporan Hasil Analisa

Gambar 4.10 Use Case Laporan Hasil Analisa

d. Analisa Fungsional Sistem

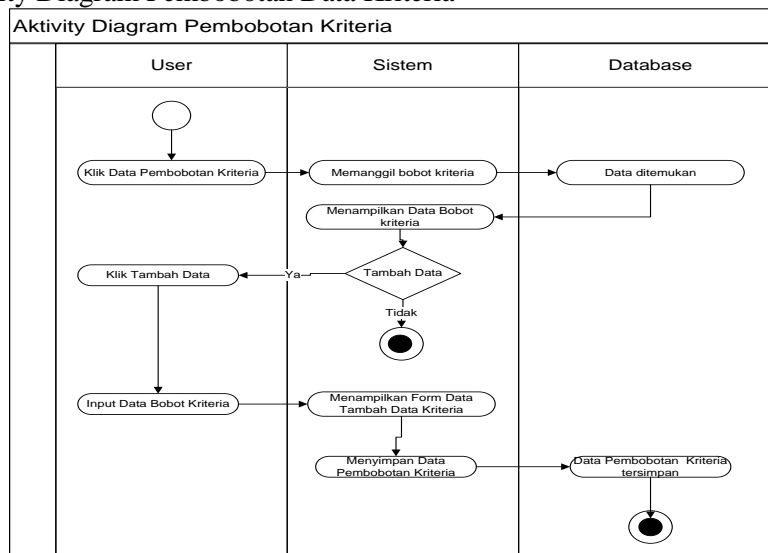
Tahap ini adalah tahap mendefinisikan dan membuat detail dari setiap fungsi-fungsi sistem. Fungsi-fungsi sistem sudah terdefiniskan pada *use cases diagram*, sehingga yang dilakukan pada tahap ini adalah merincikan setiap *use case* yang ada dengan bantuan *activity diagram*.

1. Activity Diagram Data Karyawan



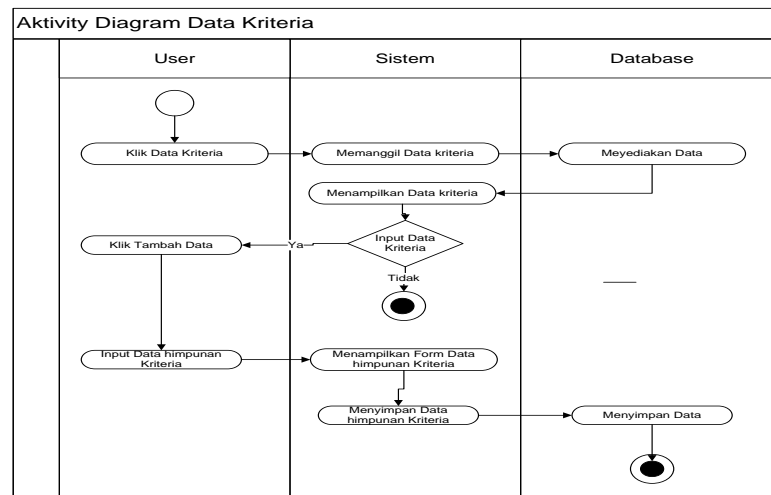
Gambar 10 Activity Diagram Data Karyawan

2. Activity Diagram Pembobotan Data Kriteria



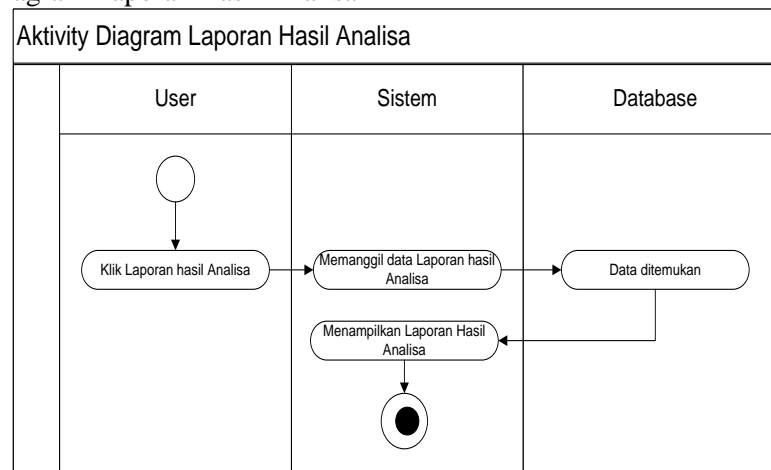
Gambar 11 Activity Diagram Pembobotan Data Kriteria

3. Activity Diagram Data Kriteria



Gambar 12 Activity Diagram Data Kriteria

4. Activity Diagram Laporan Hasil Analisa

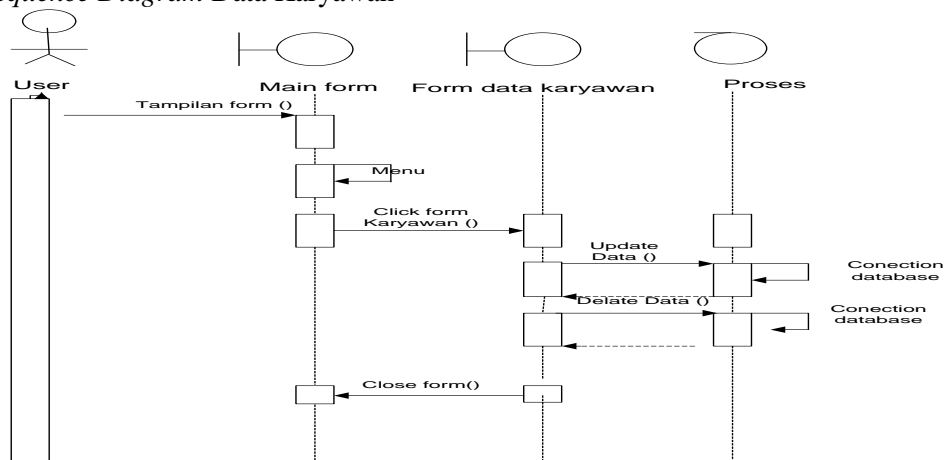


Gambar 13 Activity Diagram Laporan Hasil Analisa

e. Analisa interaksi sistem

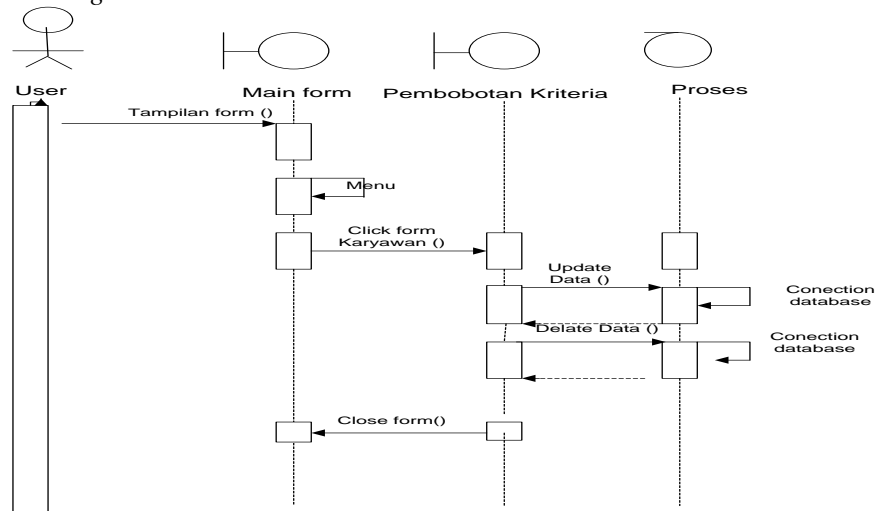
Sequence diagram berguna dalam menggambarkan bentuk interaksi antara objek sistem pendukung keputusan pemilihan karyawan terbaik.

1. Sequence Diagram Data Karyawan



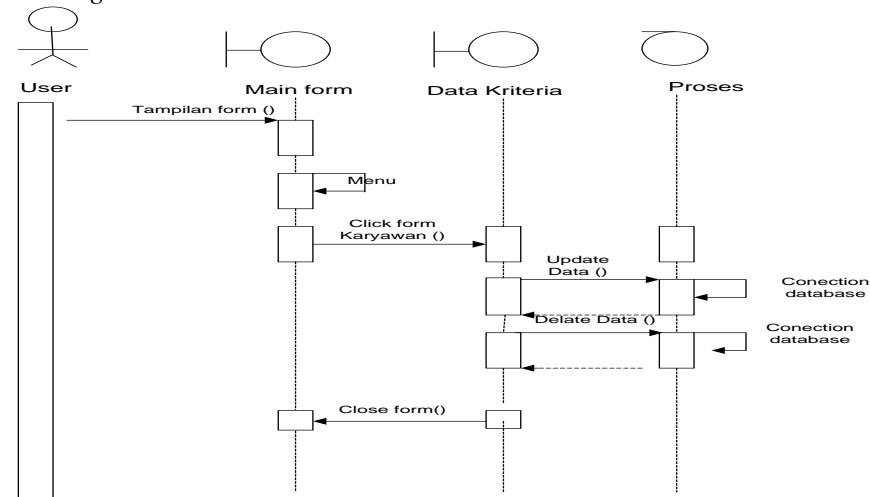
Gambar 14 Sequence Diagram Menu Data Karyawan

2. Sequence Diagram Menu Pembobotan Kriteria



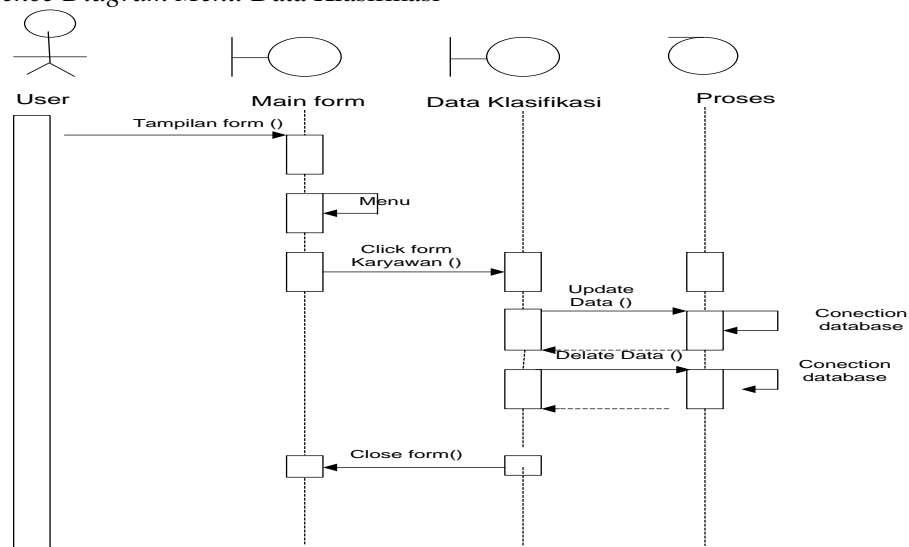
Gambar 15 Sequence Diagram Menu Pembobotan Kriteria

3. Sequence Diagram Menu Data Kriteria



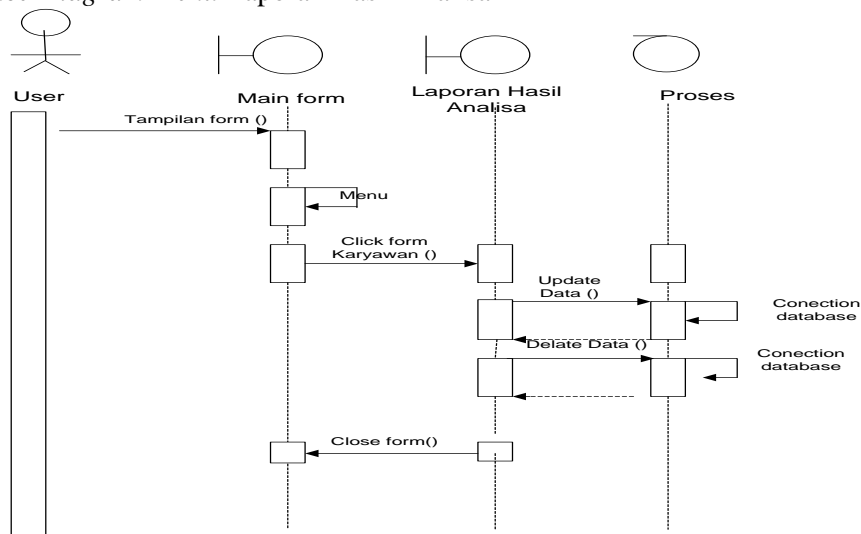
Gambar 16 Sequence Diagram Menu Data Kriteria

4. Sequence Diagram Menu Data Klasifikasi



Gambar 4.17 Sequence Diagram Menu Data Klasifikasi

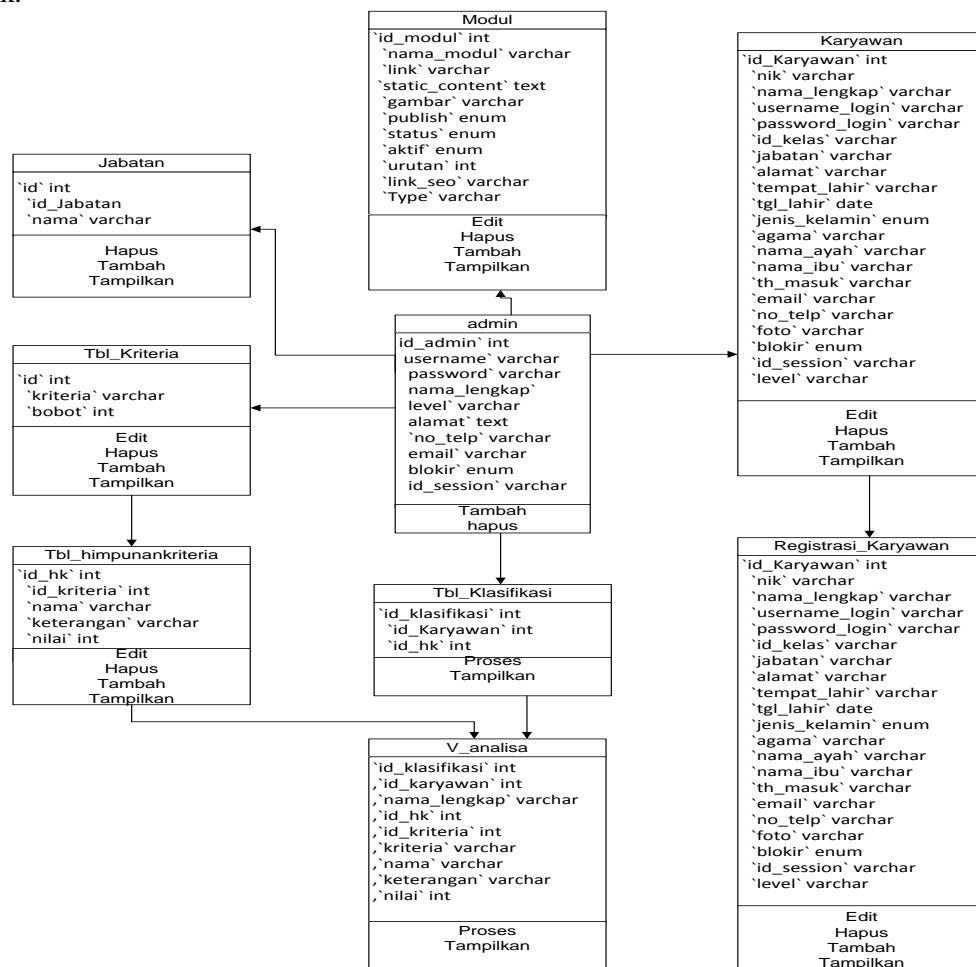
5. Sequence Diagram Menu Laporan Hasil Analisa



Gambar 18 Sequence Diagram Menu Laporan Hasil Analisa

6. Class Diagram

Class Diagram merupakan *diagram* yang menggambarkan rincian *database*, rincian tabel dan kardinalitasnya serta rincian *method* yang digunakan pada sistem pemilihan karyawan terbaik.



Gambar 19 Class Diagram

f. Rancangan Desain Sistem

Berikut Rancangan tampilan tiap-tiap Menu pada sistem pendukung keputusan karyawan terbaik pada lamoist layers batam.

a. Rancangan *Form Login*

Gambar 20 Rancangan *Form login*b. Rancangan *Main Form*

Gambar 21 Rancangan *Main Form*

c. Rancangan Form Data Karyawan

no	Nama	Aksi
1	Siti	Hapus Tambah
2	Caca	Hapus Tambah
3	Rahma	Hapus Tambah
4	Sri	Hapus Tambah
5	Elsa	Hapus Tambah

Gambar 22 Rancangan Form Data Karyawan

d. Rancangan Form Data Kriteria

Gambar 23 Rancangan Form Data Kriteria

e. Rancangan Form Data Klasifikasi

Gambar 24 Rancangan Form Data klasifikasi

f. Rancangan Form Laporan Hasil Analisa

Gambar 25 Rancangan Form Laporan Hasil Analisa

3.3.3 Implementation

Pengkodean (Coding)

Dalam pembuatan program (*coding*) peneliti menggunakan PHP versi 5.6.3 dengan kode untuk mendapatkan hasil akhir sebagai berikut:

```
$klasifikasi = mysql_query("SELECT * FROM v_analisa WHERE id_karyawan = 'R[id_siswa]"); while ($n=mysql_fetch_array($klasifikasi))
$scmax= mysql_fetch_array(mysql_query("SELECT max(nilai) as nilaimax FROM v_analisa WHERE id_kriteria='$n[id_kriteria]"));
$shimpunankriteria=mysql_fetch_array(mysql_query("SELECT*FROM tbl_himpunankriteria WHERE id_hk='$n[id_hk]"));
$nilaiok = $shimpunankriteria['nilai'] / $scmax['nilaimax'];
```

Basis data menggunakan *MYSQL*, Xampp versi 5.6.3 sebagai web servernya, untuk pengolahan tampilannya peneliti menggunakan *Bootstrap*. Desain tools peneliti menggunakan *Microsoft Office Visio 2007* dan *Microsoft Word 2007* dalam pembuatan teks laporan, emulator yang digunakan sebagai browser menggunakan *Mozilla Firefox* sebagai web browsernya.

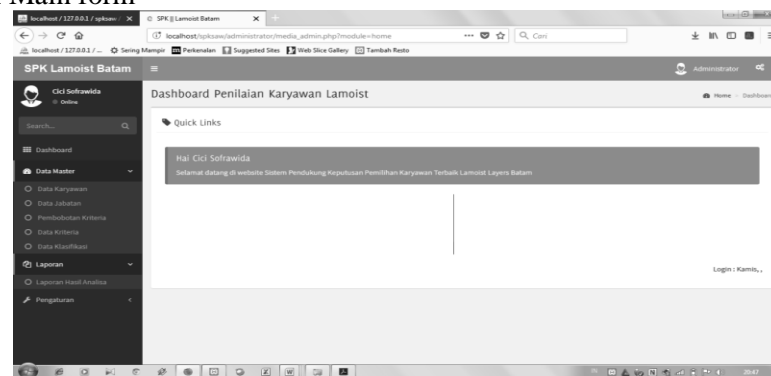
a. Hasil Implementasi

Berikut ada hasil implementasi program sistem pendukung keputusan pemilihan karyawan terbaik lamoist layers batam.

a. Tampilan Halaman Login User

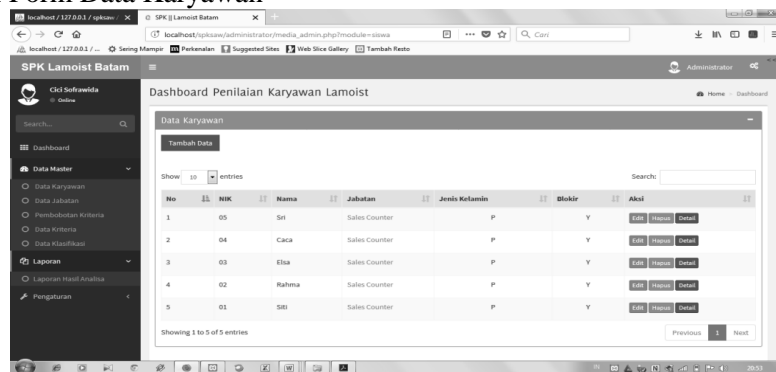
Gambar 26 Tampilan Halaman Login User

b. Tampilan Main form



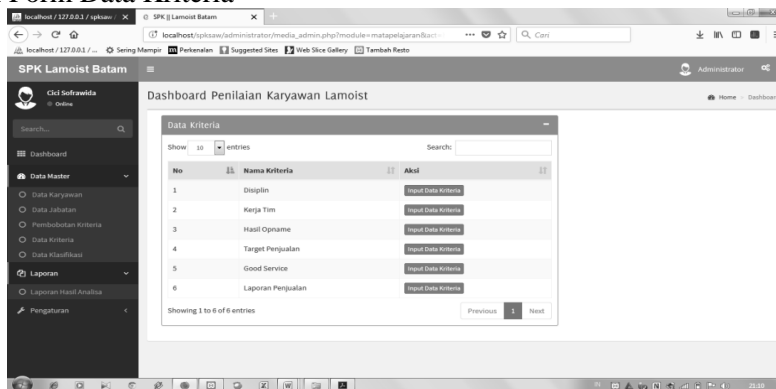
Gambar 27 Tampilan Main Form

c. Tampilan Form Data Karyawan



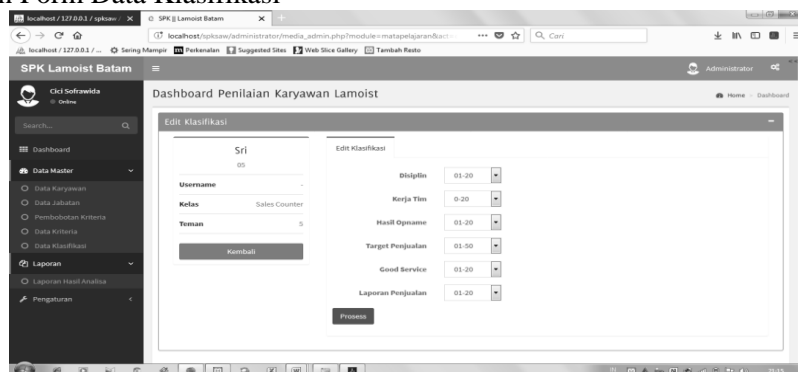
Gambar 28 Tampilan Form Data Karyawan

d. Tampilan Form Data Kriteria



Gambar 29 Tampilan Form Data Kriteria

e. Tampilan Form Data Klasifikasi



Gambar 30 Tampilan Data Klasifikasi

f. Tampilan Form Laporan Hasil Analisa

The screenshot displays the 'Dashboard Penilaian Karyawan Lamoist' interface. It features a sidebar menu on the left with options like 'Data Master', 'Data Karyawan', 'Data Jabatan', 'Pembobotan Kriteria', 'Data Kriteria', 'Data Klasifikasi', and 'Laporan'. The main content area shows three tables:

Matrik Awal

No	NIK	Nama	C1	C2	C3	C4	C5	C6
1	01	Siti	61-80	81-100	21-50	01-50	41-60	21-50
2	02	Rahma	41-60	41-60	01-20	01-50	41-60	21-50
3	03	Elsa	81-100	81-100	51-100	51-100	81-100	51-100
4	04	Caca	61-80	41-60	51-100	01-50	61-80	51-100
5	05	Sri	41-60	61-80	21-50	01-50	21-40	21-50

Normalisasi

No	NIK	Nama	C1	C2	C3	C4	C5	C6
1	01	Siti	0.8	1	0.5	0.5	0.6	0.5
2	02	Rahma	0.6	0.6	0.2	0.5	0.6	0.5
3	03	Elsa	1	1	1	1	1	1
4	04	Caca	0.8	0.6	1	0.5	0.8	1
5	05	Sri	0.6	0.8	0.5	0.5	0.4	0.5

Ranking

No	NIK	Nama	Total Nilai
1	01	Siti	62
2	02	Rahma	51
3	03	Elsa	100
4	04	Caca	75
5	05	Sri	54

Gambar 31 Tampilan Form Laporan Hasil Analisa

3.3.4 Pengujian (Testing)

Pada tahapan pengujian ini menggunakan metode *black box testing* yaitu pengujian yang berfokus pada persyaratan fungsional. Dalam peneliti ini peneliti menggunakan spesifikasi perangkat keras dan lunak antara lain :

1. Perangkat Keras

Untuk menjalankan Sistem dibutuhkan perangkat keras (hardware) yang mampu mendukung pengoperasian program tersebut. Adapun spesifikasi optimal hardware yang dibutuhkan sistem adalah :

- Laptop Lenovo core i3 64 bit
- Keyboard
- Monitor
- Mouse

2. Perangkat Lunak

Untuk menerapkan sistem, perangkat lunak (software) yang dibutuhkan tidak terlalu banyak. Adapun software yang dibutuhkan oleh sistem baru adalah :

- Sistem operasi yang digunakan adalah windows 7.
- Web browser yang digunakan adalah Mozilla Firefox dan Google Chrome.
- Aplikasi pendukung lainnya *Xampp* dan *notepad ++*.

- d. Argo UML untuk membuat rancangan umlnya.
- e. Microsoft visio untuk membuat tampilan rancangan sistem.

Uji coba perangkat lunak sistem pendukung keputusan ini dilakukan dengan cara ujicoba fungsional terhadap komponen-komponen serta fitur sistem pendukung keputusan, selain itu ujicoba dilakukan oleh narasumber untuk meminta masukan dan saran.

Tabel 9 Hasil Pengujian Sistem

No	Menu	Test yang dilakukan	Hasil yang diharapkan	Alamat	Hasil Akhir
1	Login	Melakukan login ke sistem dengan menggunakan username dan password	Berhasil masuk ke sistem	http://localhost/spksaw/administrator/	OK
2	Data Karyawan	Memanggil data karyawan dan melakukan tambah data	data di temukan dan berhasil ditambahkan	http://localhost/spksaw/administrator/media_admin.php?module=karyawan	OK
3	Data Jabatan	Memanggil data Jabatan	data di temukan	http://localhost/spksaw/administrator/media_admin.php?module=Jabatan	OK
4	Pembobotan Kriteria	Melakukan tambah data, edit, hapus pada setiap himpunan kriteria	Berhasil melakukan tambah, edit dan hapus data himpunan kriteria	http://localhost/spksaw/administrator/media_admin.php?module=kriteria	OK
5	Data Kriteria	Melakukan tambah data Kriteria, edit kriteria dan hapus kriteria ke sistem	Berhasil melakukan tambah, edit dan hapus data kriteria	http://localhost/spksaw/administrator/media_admin.php?module=kriteria&act=himpunankriteria	OK
6	Data Klasifikasi	Melakukan input nilai klasifikasi himpunan kriteria pada setiap nama karyawan ke sistem	Nilai klasifikasi berhasil di input	http://localhost/spksaw/administrator/media_admin.php?module=kriteria&act=klasifikasi	OK
7	Laporan Hasil Analisa	Memanggil hasil Penilaian karyawan	Hasil penilaian berhasil ditampilkan	http://localhost/spksaw/administrator/media_admin.php?module=kriterian&act=analisa	OK

Tabel 9 diatas menjelaskan hasil pengujian fungsional terhadap sistem perangkat lunak pendukung keputusan pemilihan karyawan terbaik pada lamoist layers batam. Hasil test menunjukkan semua komponen berjalan dengan baik.

4. KESIMPILAN

Dari hasil penelitian yang telah dilakukan maka dapat ditarik kesimpulan dalam penelitian ini adalah sebagai berikut :

1. Untuk menerapkan metode *simple additive weighting* kedalam sistem pendukung keputusan pemilihan karyawan terbaik pada lamoist layers batam terlebih dahulu penulis melakukan wawancara dengan manajemen lamoist layers batam untuk mendapatkan nilai kriteria penilaian (Ci) dan bobot kriteria sebagai berikut C1: Disiplin (20), C2: Kerja Tim (10), C3: Hasil Opname (10), C4: Target Penjualan (20), C5: *Good Service* (10) dan C6:

- Laporan Penjualan (10) sehingga dapat dilakukan perkalian matrik alternatif dan normalisasi matrik sehingga diperoleh alternatif V1 sebagai karyawan terbaik dengan nilai 100 untuk kemudian di implementasikan ke dalam sistem.
2. Untuk merancang sistem pendukung keputusan karyawan terbaik penulis terlebih dahulu melakukan pengumpulan data, melakukan analisa terhadap sistem berjalan, merancang sistem yang diusulkan, membuat design sistem serta melakukan implementasi sistem kedalam bahasa pemrograman php.

5. SARAN

Berdasarkan hasil penelitian ini ada beberapa saran yang penulis usulkan untuk mengembangkan sistem agar menjadi lebih baik, diantaranya sebagai berikut:

1. Sistem pendukung keputusan pemilihan karyawan terbaik ini hanya sebagai alat bantu manajemen dalam melakukan pemilihan karyawan terbaik hasil akhir tetap ada pada *Owner* selalu manajemen tertinggi pada lamoist layers batam.
2. Sistem ini tidak dapat melakukan *export* dan hanya dapat menyimpan data penelitian untuk satu kali periode, diharapkan untuk peneliti selanjut dapat mengembangkan program dan menambahkan fitur-fitur yang belum ada.

DAFTAR PUSTAKA

- Adam W& Rohaendi A (2017).Sistem Pendukung Keputusan Pemilihan Karyawan Terbaik Menggunakan Metode *Simple Additive Weighting* (Saw) Di Pt. Eltran Indonesia , 5(1), 6313046.
- Andi Muh. Lukman. (2016). Perancangan Sistem Informasi Pemasaran Dan Penjualan Berbasis *Smartphone* (Android) Pada Depot Air Minum, 8(1), 44–48H
- Anto A. G, Mustafidah & Suyadi A (2015). Sistem Pendukung Keputusan Penilaian Kinerja Menggunakan Metode Saw (*Simple Additive Weighting*) Di Universitas Purwokerto, 8(1),193-200.
- Aprianti, W., & Maliha, U. (2017). Sistem Informasi Kepadatan Penduduk Kelurahan Atau Desa Studi Kasus Pada Kecamatan Bati-Bati Kabupaten Tanah Laut. Jurnal Sains Dan Informatika, 2(1).
- A.S, R., & Shalahuddin, M. (2011). Rekayasa Perangkat Lunak. Modula. Bandung.
- Daerah, P. S. K. P. Analisis Dan Pengembangan Aplikasi Pembayaran Studi Kasus Perusahaan Daerah Air Minum (Pdam) Tirta Prabujaya Kota Prabumulih.
- Ependi, U. (2014). Implementasi Metode Ooad Pada Perancangan Kamus Istilah Akuntansi Berbasis Mobile. In Seminar Nasional Teknologi Informasi Dan Komunikasi 2014 (Sentika 2014).
- Fitri, N. Y., & Nurhadi, N. (2017). Analisis Dan Perancangan Sistem Pendukung Keputusan Penilaian Kinerja Guru Dengan Menggunakan Metode *Simple Additive Weighting* (Saw) Pada Smk Yadika Jambi. Jurnal Manajemen Sistem Informasi, 2(1), 318-326.
- Gunawan, I. Magister Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia Yogyakarta.

- Haryanti, S. S. (2015). Rancang Bangun Sistem Informasi E-Commerce Untuk Usaha Fashion Studi Kasus Omah Mode Kudus. *Speed-Sentra Penelitian Engineering dan Edukasi*, 3(1).
- Hasugian, P. S., Hutahaean, H. D., & Sihotang, H. T. (2017). Sistem Pendukung Keputusan Penentuan Guru Wali Kelas Pada Smp Negeri 19 Medan Dengan Menggunakan Metode *Simple Additive Weighting*. *Journal Of Informatic Pelita Nusantara*, 2(1).
- Husda, N. E. (2013). Pengantar Teknologi Informasi. (T. Wangdra & Thiorida,Eds.), Baduose Media Jakarta (1st Ed.). Jakarta: Baduose Media Jakarta.
- Ladjamudin, B. A. 2005. Analisis dan Desain Sistem Informasi. Yogyakarta: GRAHA ILMU.
- Prasetyotomo, Azis Senoaji. (2015). Pengembangan sistem informasi geospasial berbasis mobile pada perlindungan tenaga kerja Indonesia: studi kasus BNP2TKI.
- Primahudi, A. B., Suciono, F. A., & Widodo, A. A. (2016). Sistem Pendukung Keputusan Untuk Pemilihan Karyawan Dengan Metode *Simple Additive Weighting* Di Pt. Herba Penawar Alwahida Indonesia. *Jimp-Jurnal Informatika Merdeka Pasuruan E-Issn 2503-1945*, 1(2).
- Ropianto, M. (2016). Pemahaman Penggunaan *Unified Modelling Language*. *Jurnal Teknik Ibnu Sina Jt-Ibsi*, 1(01).
- Safitri, K., Waruwu, F. T., & Mesran, M. (2017). Sistem Pendukung Keputusan Pemilihan Karyawan Berprestasi Dengan Menggunakan Metode *Analytical Hieararchy Process* (Studi Kasus: Pt. Capella Dinamik Nusantara Takengon). *Media Informatika Budidarma*, 1(1).
- Setyabudhi, A. L. (2017). Perancangan Sistem Informasi Pengolahan Data Absensi dan Pengambilan Surat Cuti Kerja Berbasis Web. *JR: JURNAL RESPONSIVE Teknik Informatika*, 1(1).
- Setiaji, P., 2016. Sistem Pendukung Keputusan Dengan Metode *Simple Additive Weighting*. *Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, 1(1), pp.59-67.
- Subawa, I. G. B., Wirawan, I. M. A., & Sunarya, I. M. G. (2015). Pengembangan Sistem Pendukung Keputusan Pemilihan Pegawai Terbaik Menggunakan Metode *Simple Additive Weighting (Saw)* Di Pt Tirta Jaya Abadi Singaraja. *Karmapati (Kumpulan Artikel Mahasiswa Pendidikan Teknik Informatika)* Issn: 2252-9063, 4(5).
- Suharti. (2017) Daerah, P. S. K. P. Analisis Dan Pengembangan Aplikasi Pembayaran Studi Kasus Perusahaan Daerah Air Minum (Pdam) Tirta Prabujaya Kota Prabumulih.
- Triwayuni A, Septiawan M. R, Rizal, & Marsusilanti (2015).Sistem Pendukung Keputusan Penentuan Karyawan Terbaik *Carrefour* Menggunakan Metode *Simple Additive Weighting (Saw)*, 15(1),66-80.
- Wiyono S. E & Latipah (2017). Penerapan Metode *Simple Additive Weighting (Saw)* Pada Sakinah Supermaket Untuk Pemilihan Karyawan Terbaik, 5(1), 6313046.