

**Perancangan Sistem Penilaian Karyawan Menggunakan
Framework Laravel Berbasis *Simple Additive Weighting* (Studi Kasus PT
Pura Barutama Unit *Paper Mill* 5, 6, 9)**

Artikel Ilmiah



Peneliti:

**Andreas Anthapillar Vega Auxyllian (672015096)
Yeremia Alfa Susetyo, S.Kom., M.Cs.**

**Program Studi S1 Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Salatiga
2019**

**Perancangan Sistem Penilaian Karyawan Menggunakan
Framework Laravel Berbasis *Simple Additive Weighting* (Studi Kasus PT
Pura Barutama Unit *Paper Mill* 5, 6, 9)**

Artikel Ilmiah

**Diajukan kepada
Fakultas Teknologi Informasi
Untuk memperoleh Gelar Sarjana Komputer**



**Peneliti:
Andreas Anthapillar Vega Auxyllian (672015096)
Yeremia Alfa Susetyo, S.Kom., M.Cs.**

**Program Studi S1 Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Salatiga
2019**



PERNYATAAN TIDAK PLAGIAT

Saya yang bertanda tangan di bawah ini:

Nama : Andreas Anthapillar Vega Auxyllian
NIM : 672015096 Email : pillar_vega@aol.com
Fakultas : Fakultas Teknologi Informasi Program Studi : Teknik Informatika
Judul tugas akhir : Perancangan Sistem Penilaian Karyawan Menggunakan Framework Laravel Berbasis Simple Additive Weighting (Studi Kasus PT Pura Barutama Unit Paper Mill 5, 6, 9)
Pembimbing : Yermia Alfa Susetyo, S.Kom, M.Cs.

Dengan ini menyatakan bahwa:

1. Hasil karya yang saya serahkan ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar keserjanaan baik di Universitas Kristen Satya Wacana maupun di institusi pendidikan lainnya.
2. Hasil karya saya ini bukan saduran/terjemahan melainkan merupakan gagasan, rumusan, dan hasil pelaksanaan penelitian/implementasi saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing akademik dan narasumber penelitian.
3. Hasil karya saya ini merupakan hasil revisi terakhir setelah diujikan yang telah diketahui dan disetujui oleh pembimbing.
4. Dalam karya saya ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali yang digunakan sebagai acuan dalam naskah dengan menyebutkan nama pengarang dan dicantumkan dalam daftar pustaka.

Pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari terbukti ada penyimpangan dan ketidakbenaran dalam pernyataan ini maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya saya ini, serta sanksi lain yang sesuai dengan ketentuan yang berlaku di Universitas Kristen Satya Wacana.

Salatiga, 1 AGUSTUS 2019



Andreas Anthapillar Vega Auxyllian

Tanda tangan & nama terang mahasiswa



PERNYATAAN PERSETUJUAN AKSES

Saya yang bertanda tangan di bawah ini:

Nama : Andreas Anthapillar Vega Auxyllian
NIM : 672015096 Email : pillar_vega@aol.com
Fakultas : Fakultas Teknologi Informasi Program Studi : Teknik Informatika
Judul tugas akhir : Perancangan Sistem Penilaian Karyawan Menggunakan Framework Laravel Berbasis Simple Additive Weighting (Studi Kasus PT Pura Barutama Unit Paper Mill 5, 6, 9)

Dengan ini saya menyerahkan hak *non-eksklusif** kepada Perpustakaan Universitas – Universitas Kristen Satya Wacana untuk menyimpan, mengatur akses serta melakukan pengelolaan terhadap karya saya ini dengan mengacu pada ketentuan akses tugas akhir elektronik sebagai berikut (beri tanda pada kotak yang sesuai):

- ☒ a. Saya mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA
- ☐ b. Saya tidak mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA**

* Hak yang tidak terbatas hanya bagi satu pihak saja. Pengajar, peneliti, dan mahasiswa yang menyerahkan hak non-eksklusif kepada Repositori Perpustakaan Universitas saat mengumpulkan hasil karya mereka masih memiliki hak copyright atas karya tersebut.

** Hanya akan menampilkan halaman judul dan abstrak. Pilihan ini harus dilampiri dengan penjelasan/ alasan tertulis dari pembimbing TA dan diketahui oleh pimpinan fakultas (dekan/kaprodi).

Demikian pernyataan ini saya buat dengan sebenarnya.

Salatiga, 1 Agustus 2019

Andreas Anthapillar Vega Auxyllian

Tanda tangan & nama terang mahasiswa

Mengetahui,

Yerima Alfa Susetyo, S.Kom, M.Cs.

Tanda tangan & nama terang pembimbing

Lembar Persetujuan

Perancangan Sistem Penilaian Karyawan Menggunakan *Framework Laravel* Berbasis *Simple Additive Weighting* (Studi Kasus PT Pura Barutama Unit Paper Mill 5, 6, 9)

Artikel Ilmiah

Peneliti :

Andreas Anthapillar Vega Auxyllian (672015096)

Telah disetujui untuk diuji:

Tanggal : 11 Juli 2019



Yeremia Alfa Susetvo, S.Kom., M.Cs.

Pembimbing

Lembar Pengesahan

Judul Tugas Akhir : Perancangan Sistem Penilaian Karyawan Menggunakan
Framework Laravel Berbasis *Simple Additive Weighting*
(Studi Kasus PT Pura Barutama Unit *Paper Mill* 5, 6, 9)
Nama Mahasiswa : ANDREAS ANTHAPILAR VEGA AUXYLLIAN
NIM : 672015096
Program Studi : Teknik Informatika
Fakultas : Teknologi Informasi

Menyetujui,

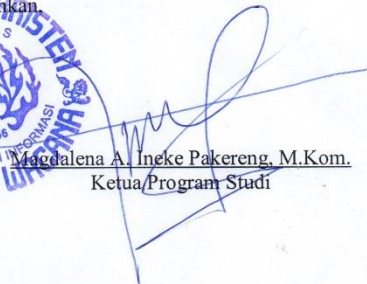


Yerima Alfa Susetyo, S.Kom., M.Cs.
Pembimbing

Mengesahkan



Dr. Wiwin Sulisty, S.T., M.Kom.
Dekan



Magdalena A. Ineke Pakereng, M.Kom.
Ketua Program Studi

Dinyatakan Lulus Tanggal: 19 Juli 2019

Reviewer :

- Evangs Mailoa, S.Kom., M.Cs.



1. Pendahuluan

Penilaian Kinerja atau *Performance Appraisal (PA)* adalah sebuah sistem yang formal digunakan dalam beberapa periode waktu tertentu untuk menilai prestasi kerja seorang karyawan (Setiobudi, 2017). Selain itu, penilaian kinerja dapat berfungsi untuk mengidentifikasi, mengobservasi, mengukur, mendata, serta melihat kekuatan dan kelemahan dari karyawan dalam melakukan pekerjaan (Setiobudi, 2017). PURA berawal dari sebuah percetakan letterpress kecil yang berdiri di tahun 1908 (Pura Group, 2018). Kemudian di tahun 1970, di bawah kepemimpinan generasi ketiga Jacobus Busono perusahaan berkembang pesat sampai menjadi industri modern Pura Group yang kita kenal saat ini (Pura Group, 2018). PT. Pura Barutama Unit Paper Mill 5, 6, 9 merupakan salah satu anak perusahaan PT. Pura Group yang bergerak pada bidang industri kertas (Chusna, Huboyo, & Andarani, 2017). Unit tersebut memiliki jumlah karyawan mencapai lebih kurang 500 karyawan menyebabkan perusahaan kurang memantau kinerja karyawan.

Penilaian karyawan yang dijalankan saat ini adalah dengan menyebarkan *form* penilaian karyawan. Setiap karyawan dinilai oleh pimpinan yang menaungi departemen-departemen yang ada dengan mengisi *form* penilaian karyawan. Hasil data kemudian diolah melalui perhitungan manual yang dilakukan menggunakan aplikasi pengolah angka. Penilaian ini membutuhkan waktu yang lama tergantung dari berapa banyak orang yang dinilai pada setiap bidang. Kurangnya hasil dari penilaian turut dalam permasalahan disebabkan penilaian yang dilakukan saat ini cenderung memihak serta bersifat subjektif.

Terdapat beberapa metode *fuzzy* untuk menyelesaikan permasalahan penilaian prestasi karyawan, antara lain dengan menggunakan *Multi Attribute Decision Making (MADM)* dengan metode *Simple Additive Weighting (SAW)*, *Weighted Product (WP)*, *ELECTRE*, *TOPSIS*, *Analytic Hierarchy Process (AHP)* (Nufus, 2016). Metode *Simple Additive Weighting* merupakan salah satu metode pendukung keputusan yang mencari penjumlahan terbobot dari *rating* kinerja pada setiap alternatif pada semua atribut (Masri, 2016). Metode *Simple Additive Weighting* lebih cocok diterapkan pada perusahaan yang telah menetapkan pembobotan kriterianya serta skala penilaiannya dan lebih mengutamakan kemudahan dalam implementasi (Shiddieq & Septyan, 2017).

Dibutuhkan sebuah *framework* pemrograman yang dapat menghitung data dalam menunjang sebuah sistem pendukung keputusan. Sistem ini dibawa dengan menggunakan pemrograman PHP serta *framework Laravel* sebagai pilihan. *Laravel* merupakan *framework* yang fokus pada alur pembelajaran yang mudah, sehingga memangkas waktu dari pembuatan aplikasi hingga penerbitan aplikasi (Stauffer, 2017). Semua pekerjaan dasar yang dilakukan oleh sebuah aplikasi *web* dapat disederhanakan melalui komponen-komponen yang telah disediakan oleh *Laravel* (Stauffer, 2017).

Berdasarkan kasus yang ada serta teknologi yang ada, dapat dirumuskan beberapa masalah antara lain cara merancang sistem penilaian karyawan, penerapan metode *Simple Additive Weighting* dalam sistem penilaian karyawan dan cara kerja *framework Laravel* dalam membangun sistem penilaian karyawan. Tujuan dalam penelitian ini adalah untuk merancang dan menerapkan sistem penilaian karyawan menggunakan metode *Simple Additive Weighting* dan *framework Laravel*. Adapun manfaat yang didapat dari penelitian ini adalah untuk memudahkan perusahaan dalam menjalankan program penilaian karyawan dan departemen *HR-GA* unit *Paper Mill 5, 6, 9* dapat mengetahui serta memberikan keputusan

kepada karyawan yang memiliki prestasi atau karyawan yang memiliki performa buruk terhadap pekerjaan.

2. Tinjauan Pustaka

Penelitian ini dilakukan berdasarkan beberapa penelitian terdahulu. Ada lima penelitian terdahulu yang sesuai dengan topik pembahasan yaitu metode *Simple Additive Weighting* dan framework *Laravel*. Penelitian yang pertama membahas tentang keunggulan *Laravel* yang dapat diterapkan pada sistem penilaian kinerja karyawan pada PT. Sejahtera Motor Gemilang. Penelitian tersebut menghasilkan sebuah aplikasi yang telah memenuhi kebutuhan perusahaan dalam proses pengukuran kinerja karyawan, serta kemudahan yang didapat dari *framework Laravel* dalam proses pembuatan aplikasi (Hendrian, Devi, & Kurniawan, 2015). Perbandingan yang terdapat pada penelitian pertama adalah penggunaan metode KPI dalam pembuatan sistem dengan *framework Laravel*. Penelitian pertama menggunakan metodologi *scrum* sehingga perancangan dibagi berdasarkan prioritas sedangkan dalam penelitian ini perancangan aplikasi dilakukan secara menyeluruh dari segi perancangan.

Penelitian kedua diambil dari jurnal yang membahas tentang implementasi *framework Laravel* pada perekrutan pelamar di *Godrej* Indonesia. *Framework Laravel* dipilih dengan alasan tersedianya alat, komponen, dan desain yang mempercepat pembangunan sebuah *framework* (Synta, Ginanjar, Juwono, Hadisukmana, & Wahyu, 2017). Penelitian ini memiliki perbedaan dengan penelitian yang dilakukan antara lain dari tujuan pembuatan implementasi *framework* adalah untuk menilai pantas atau tidaknya seorang pelamar dalam pekerjaan.

Penelitian selanjutnya merupakan kasus yang diselesaikan dengan menggunakan metode *simple additive weighting*. Metode *simple additive weighting* dipilih karena dapat memilih alternatif yang terbaik. Metode *simple additive weighting* juga membantu manajer untuk membuat keputusan yang cepat dan tepat, sehingga pada penelitian yang dilakukan oleh Nashrudin Setiawan, dkk metode *simple additive weighting* dipilih sebagai metode penilaian karyawan dalam menentukan tunjangan pegawai (Setiawan et al., 2018). Penelitian ketiga tersebut merupakan penelitian dengan memanfaatkan *simple additive weighting* untuk penentuan tunjangan pegawai, sedangkan penelitian ini memanfaatkan *simple additive weighting* untuk menentukan peringkat karyawan terbaik setiap departemen.

Metode ini juga dilakukan pada penelitian perbandingan penilaian karyawan dengan menggunakan metode *simple additive weighting* terhadap *composite performance index* untuk penentuan tunjangan pegawai di PT Sepatu Mas Idaman. Dibandingkan dengan metode *composite performance index*, diketahui bahwa *simple additive weighting* memiliki efisiensi sebesar 80 hingga 93% (Karlitasari, Suhartini, & Benny, 2017). Penelitian keempat merupakan penelitian untuk membandingkan penilaian karyawan dengan menggunakan *simple additive weighting* terhadap *composite performance index*. Penelitian keempat menjadi acuan dari penelitian ini serta membuktikan penggunaan *simple additive weighting* merupakan metode yang efisien dilihat dari nilai efisiensi sebesar 80 hingga 93 persen.

Penelitian berikutnya merupakan pemanfaatan metode *simple additive weighting* dalam seleksi pelamar daring dengan kombinasi berupa logika *Tahani fuzzy*. Tujuan dari penelitian tersebut adalah untuk mengoptimalkan kinerja dari perusahaan dalam merekrut karyawan baru. Hasil dari penelitian tersebut adalah peringkat kandidat pelamar, pekerjaan ideal pelamar, serta hasil akhir peringkat (Suprpty, Malani, & Nurhayati, 2016). Perbandingan pada penelitian ini terletak pada metode yang diberikan. Metode yang digunakan pada penelitian ini hanya *simple*

additive weighting saja, sedangkan pada penelitian terakhir menggunakan kombinasi metode fuzzy Tahani.

Simple Additive Weighting

Menurut Fishburn dan MacCrimmon (dalam Frieyadie, 2016) mengemukakan bahwa metode *Simple Additive Weighting* (SAW), sering juga dikenal dengan istilah metode penjumlahan terbobot. Konsep dasar metode *Simple Additive Weighting* (SAW) adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut (Masri, 2016). Menurut (Asnawati & Kanedi, 2012) “Kriteria penilaian dapat ditentukan sendiri sesuai dengan kebutuhan perusahaan”.

Menurut Fishburn dan MacCrimmon dalam (Frieyadie, 2016) Ada beberapa langkah dalam penyelesaian metode *Simple Additive Weighting* (SAW) adalah sebagai berikut:

1. Menentukan kriteria-kriteria yang dijadikan acuan dalam pendukung keputusan yaitu C_i .
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan kriteria (C_i).
4. Kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan maupun atribut biaya) sehingga diperoleh matriks ternormalisasi R .
5. Hasil akhir diperoleh dari proses perangkingan yaitu penjumlahan dari perkalian matriks ternormalisasi R dengan vector bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatis terbaik (A_i) sebagai solusi.

Rumus untuk menghitung kriteria/ atribut dari metode *Simple Additive Weighting* adalah:

$$R_{ij} = \begin{cases} \frac{X_{ij}}{\max X_{ij}} & \text{jika } j \text{ adalah atribut (keuntungan)/ benefit} \\ \frac{\min X_{ij}}{X_{ij}} & \text{jika } j \text{ adalah atribut (biaya)/ cost} \end{cases}$$

Dimana :

R_{ij} = Rating kinerja ternormalisasi

$\max_{ij}$ = Nilai maksimum dari setiap baris dan kolom

$\min_{ij}$ = Nilai minimum dari setiap baris dan kolom

X_{ij} = Baris dan kolom dari matriks Dengan R_{ij} adalah rating kinerja ternormalisasi dari alternatif A_i pada atribut C_j ; $i=1,2,\dots,m$ dan $j=1,2,\dots,n$.

Untuk menghitung bobot alternatif terhadap bobot kriteria/ atribut dari metode Simple Additive Weighting adalah sebagai berikut.

$$V_i = \sum_{j=1}^n W_j \cdot R_{ij}$$

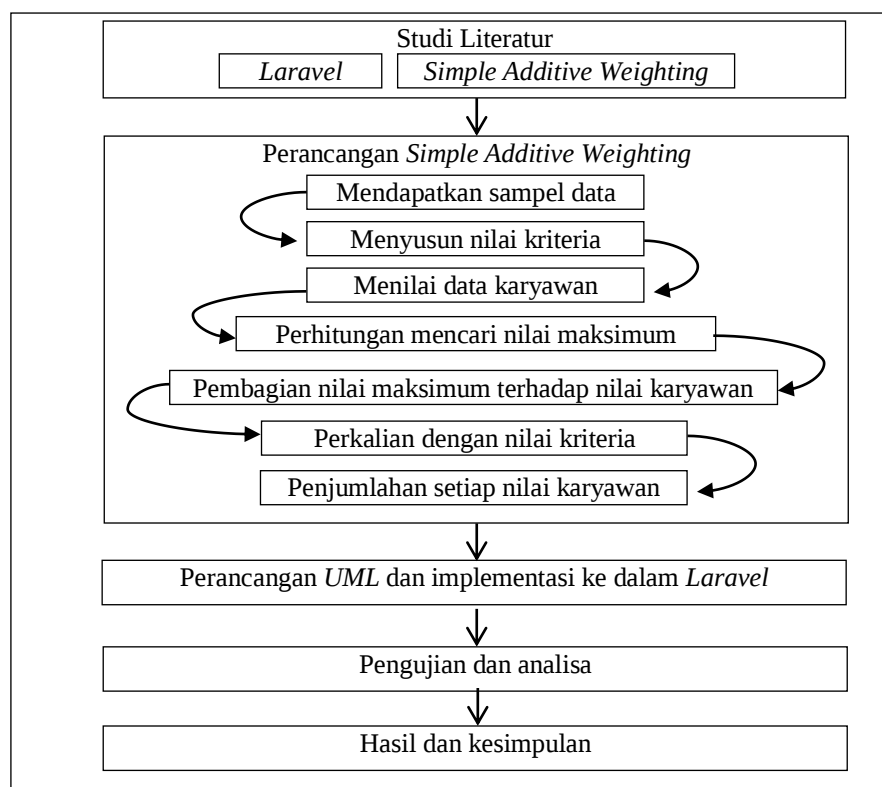
Nilai yang lebih besar mengindikasikan bahwa alternatif lebih terpilih. Nofriansyah dalam (Frieyadie, 2016) menjelaskan bahwa Metode *Simple Additive Weighting* disarankan untuk menyelesaikan masalah penyeleksian dalam sistem pengambilan keputusan multi proses. Metode *Simple Additive Weighting* (SAW) merupakan metode yang banyak digunakan dalam pengambilan keputusan yang memiliki banyak atribut.

Laravel

Laravel adalah salah satu *framework* PHP terbaik yang dikembangkan oleh Taylor Otwell, proyek *Laravel* dimulai pada April 2011 (Yudhanto & Prasetyo, 2018). Awal mula *Laravel* dibuat, karena Otwell sendiri tidak menemukan *framework* yang *up to date* dengan versi PHP (Yudhanto & Prasetyo, 2018). *Laravel* adalah sebuah *framework* PHP yang dirilis dibawah lisensi MIT dan dibangun dengan konsep MVC (*model view controller*) (Yudhanto & Prasetyo, 2018). Yang menarik dari *Laravel* adalah sintaksnya yang ekspresif, elegan, serta dirancang khusus untuk memudahkan dan mempercepat proses *web development* (Yudhanto & Prasetyo, 2018). *Laravel* merupakan *framework* yang fokus pada alur pembelajaran yang mudah, sehingga memangkas waktu dari pembuatan aplikasi hingga penerbitan aplikasi (Stauffer, 2017). Semua pekerjaan dasar yang dilakukan oleh sebuah aplikasi *web* dapat disederhanakan melalui komponen-komponen yang telah disediakan oleh *Laravel* (Stauffer, 2017).

Komponen yang terdapat pada *Laravel* yang sederhana salah satunya adalah *template engine Blade*. *Blade* merupakan *template engine* yang mudah, dan cepat dari *Laravel* (Monteiro, 2018). *Blade* tidak membutuhkan pengetahuan PHP pada kode *views* (Monteiro, 2018). *Blade* yang telah dibuat akan dijalankan oleh *framework* dan akan disimpan sampai kembali dimodifikasi (Monteiro, 2018). Selain itu terdapat fitur *Eloquent ORM* yang merupakan manajemen basis data dengan metode *active record*. *Eloquent* ini dapat memudahkan pengguna *Laravel* dalam membuat bagian *create*, *retrieve*, *update*, dan *delete* pada bagian model tanpa perlu mendeklarasikan bagian itu sendiri-sendiri (Sinha, 2017).

3. Metodologi Penelitian



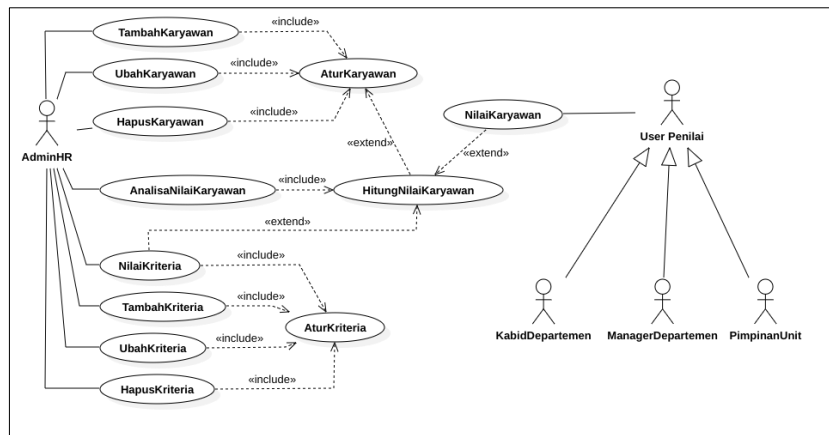
Gambar 1 Diagram Alur Perancangan Penilaian Karyawan

Tahap penelitian yang telah dirancang dan digunakan dalam sistem penilaian karyawan dengan metode *Simple Additive Weighting* sesuai dengan gambar 1 adalah:

1. Studi Literatur tentang teori pengembangan sistem penunjang keputusan dengan metode *Simple Additive Weighting* (SAW). Adapun teori yang dibutuhkan antara lain: metode *Simple Additive Weighting* (SAW), cara perhitungan metode, *framework Laravel* serta pengimplementasian *framework* ke dalam *web server*.
2. Perancangan metode *fuzzy Simple Additive Weighting* dengan mengambil sampel studi kasus pada Unit *Paper Mill* 5, 6, 9 PT Pura Barutama. Sampel data yang diambil antara lain adalah data karyawan, data departemen, serta data kriteria yang telah dibuat oleh pihak HR-GA Unit *Paper Mill* 5, 6, 9. Variabel dari data kriteria yang diberikan berupa 20 persen untuk aspek subjektif (kerjasama, kedisiplinan, K3, dan absensi), dan 80 persen untuk aspek objektif (pemahaman tugas, pencapaian, efektifitas, kreatifitas, dan ketuntasan kerja). Perhitungan metode SAW dilakukan dengan cara menyusun kriteria, menilai setiap aspek kriteria, menyusun daftar karyawan, menilai karyawan berdasarkan contoh data sampel penilaian, melakukan perhitungan mencari nilai maksimum karyawan dari setiap kriteria, membagi nilai maksimum terhadap nilai karyawan, mengalikan setiap nilai dengan nilai kriteria yang telah ditentukan, dan hasil akhirnya adalah penjumlahan dari setiap kriteria yang ada pada karyawan.
3. Perencanaan dan implementasi metode ke dalam perangkat lunak: Membuat model UML beserta *state diagram*. Perencanaan sistem juga akan menggunakan model pengembangan secara *prototype*. Pengembangan dilakukan dengan menggunakan *tools* berupa *text editor Atom*, aplikasi *web server Apache* dan basis data *MySQL*. Kerangka kerja (*framework*) pemrograman menggunakan *Laravel* dengan metode penilaian menggunakan *Simple Additive Weighting*.
4. Pengujian dan Analisa aplikasi yang telah dibuat dan analisa hasil keluaran aplikasi.
5. Pengambilan hasil serta kesimpulan diambil dengan mengambil hasil keluaran aplikasi yang telah dibuat.

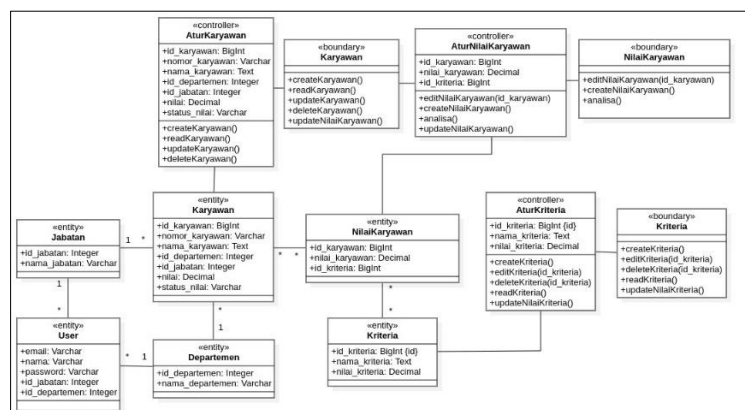
Perancangan diagram diperlukan dalam merancang dan implementasi *simple additive weighting* ke dalam *framework Laravel*. Penelitian ini membahas tiga diagram yang akan menjelaskan alur program antara lain *use case*, *class*, serta *sequence diagram*.

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Windu Gata, 2016). *Use case diagram* pada penelitian ini terdiri dari 4 aktor utama yaitu seorang admin HR, pimpinan unit, manajer, serta kepala bidang. Admin HR memiliki lebih banyak kegiatan yang dilakukan pada sistem, antara lain adalah menambahkan kriteria, mengatur dan menilai kriteria serta melakukan proses analisa terhadap nilai karyawan, sedangkan tiga aktor selain admin HR merupakan aktor yang digeneralisasikan menjadi user penilai dengan kegiatan yang sama, yaitu menilai kinerja karyawan. Berikut adalah diagram *use case* dari penelitian ini:



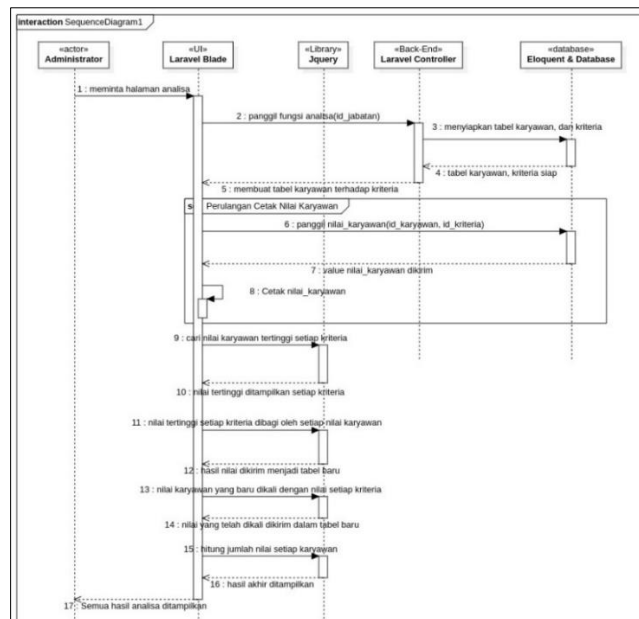
Gambar 2 Use Case Diagram

Class diagram menggambarkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem (Windu Gata, 2016). *Class diagram* pada penelitian ini dibagi menjadi *entity*, *boundary*, dan *controller*. Terdapat 6 *class entity* yang terdiri dari jabatan, departemen, karyawan, kriteria serta nilai karyawan. *Class entity* berisi semua atribut pada setiap *class*, dan setiap *class entity* memiliki relasi dengan detail relasi seperti pada gambar 1.3. *Class controller* terdapat tiga *class* yang terdiri dari *AturNilaiKriteria*, *AturKaryawan*, dan *AturKriteria*. Ketiga *controller* tersebut terhubung antara *class entity* dan *boundary* masing-masing. Kelas *boundary* terdapat *class* nilai karyawan, karyawan serta kelas kriteria yang terdiri dari operasi-operasi yang dilakukan oleh lingkungan luar (aktor sebagai perantara).



Gambar 3 Class Diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek (Windu Gata, 2016). *Sequence diagram* dalam penelitian ini diperlukan dengan tujuan untuk melihat logika yang terjadi pada bagian perhitungan analisa karyawan. Berikut adalah tahapan-tahapan dari proses perhitungan analisa karyawan:



Gambar 4 Sequence Diagram

Proses pertama yang dilakukan setelah meminta halaman analisa, adalah tampilan antarmuka berupa *file blade* dari *Laravel* akan mengirim perintah tersebut kepada *controller* nilai karyawan yang ada pada *Laravel* (proses dua). Fungsi *controller* tersebut kemudian membuka *database* serta mendapatkan data melalui fasilitas *query builder* dari *Laravel*, *Eloquent*. *Controller* tersebut juga menampilkan kembali halaman *view* untuk menyajikan data yang telah diambil dari *database* (proses lima). Terdapat dua tabel yang diambil datanya, yaitu data karyawan serta data kriteria. Halaman analisa beserta data-data yang telah diambil tampil setelah fungsi *controller* selesai. Tampilan halaman yang berupa *file blade* tersebut, kemudian ditambahkan perintah *query builder Eloquent* untuk mengambil data nilai karyawan yang ada pada *database* (proses enam). Terjadi proses *looping* untuk menampilkan nilai kriteria dari setiap karyawan dalam bentuk tabel setelah perintah *query builder* tersebut dieksekusi (proses delapan). Proses mencari nilai karyawan tertinggi dari setiap kriteria dilakukan melalui *library jQuery* dari tampilan analisa kemudian ditampilkan pada baris terbawah dalam tabel (proses sembilan dan sepuluh).

Nilai tertinggi dari setiap kriteria tersebut kemudian dibagi terhadap nilai karyawan yang telah ditampilkan tadi, lalu ditampilkan kembali dalam bentuk tabel (proses sebelas dan dua belas). Ditambahkan fungsi dari *jQuery* untuk mengalikan hasil pembagian nilai karyawan tersebut dengan nilai kriteria dalam tampilan tabel tersebut (proses tiga belas dan empat belas). Hasil dari proses tiga belas dan empat belas tersebut kemudian dilakukan penjumlahan skor nilai setiap karyawan dengan menggunakan fungsi *sum* dari *jQuery*. Hasil akhir kemudian dikirim kedalam *database* melalui tombol yang telah disediakan dalam tampilan.

4. Hasil dan Pembahasan

Perancangan sistem penilaian karyawan dilakukan sesuai dengan kondisi penilaian perusahaan pada saat penelitian ini ditulis dengan membagi proyek ke dalam 2 modul: *admin HR* sebagai *superuser*, dan user biasa yang terdiri dari manajer, kepala bidang, serta pimpinan unit. Modul admin HR memiliki 4 tugas pada penilaian yaitu mengatur data kriteria, pembobot kriteria, mengatur data karyawan, serta menganalisa hasil nilai karyawan. Modul user biasa hanya memiliki 1 tugas pada penilaian karyawan. Penilaian karyawan dibagi berdasarkan

jabatan user biasa. Kepala bidang mengerjakan penilaian pada *level* staff, kepala shift, dan operator, manajer/ kepala departemen mengerjakan penilaian pada *level* kepala bidang, dan pimpinan unit mengerjakan penilaian pada *level* manajer.

Pengujian metode *simple additive weighting* dilakukan dengan cara perhitungan manual menggunakan sampel berupa 5 karyawan, serta menggunakan 9 sampel kriteria dengan pembobotan kriteria 20 persen untuk aspek subjektif (kerjasama, kedisiplinan, K3, dan absensi), dan 80 persen untuk aspek objektif (pemahaman tugas, pencapaian, efektifitas, kreatifitas, dan ketuntasan kerja). Pembobotan kriteria dibuat berdasarkan data yang didapat dari penilaian yang ditentukan nilainya oleh perusahaan. Berikut ini adalah tabel aspek subjektif:

Tabel 1 Pembobotan Aspek Subjektif

Nama Kriteria	Alias Kriteria	Nilai
Kerjasama	C1	0.05
Kedisiplinan/ Kepatuhan	C2	0.05
K3	C3	0.05
Absensi	C4	0.05

Tabel 2 merupakan tabel aspek objektif. Terdiri dari pemahaman tugas, pencapaian hasil, efektifitas, kreatifitas, dan ketuntasan

Tabel 2 Pembobotan Aspek Objektif

Nama Kriteria	Alias Kriteria	Nilai
Pemahaman Tugas	C5	0.1
Pencapaian hasil	C6	0.25
Efektifitas	C7	0.15
Kreatifitas	C8	0.15
Ketuntasan	C9	0.15

Sampel 5 karyawan dinilai dengan nilai setiap kriteria. Bobot nilai dibuat dimulai dari angka 0,2 menunjukkan nilai kriteria tersebut sangat kurang, 0,4 menunjukkan nilai kriteria kurang, 0,6 adalah nilai kriteria cukup, 0,8 nilai kriteria baik dan 1 yang menunjukkan nilai kriteria sangat baik. Contoh pada karyawan A1 kriteria C1, memiliki bobot 1 menunjukkan bahwa karyawan A1 pada kriteria C1 dinilai sangat baik, Kriteria C2 memiliki bobot 0,8, C3 dengan bobot 0,8, C4 berbobot 0,8 dan berlanjut sampai pada kriteria C9 yang memiliki bobot 0,6. Penilaian tersebut dilakukan terus hingga karyawan A5 secara horizontal. Hasil dari setiap kriteria diambil nilai paling tinggi. Contoh pada kriteria C1, nilai yang paling tinggi diantara A1 sampai dengan A5 adalah nilai 1. Hasil dari semua proses di atas terlampir seperti pada tabel ini:

Tabel 3 Tabel pencarian nilai tertinggi

	C1	C2	C3	C4	C5	C6	C7	C8	C9
A1	1	0,8	0,8	0,8	0,8	0,8	0,6	0,6	0,6
A2	0,6	0,6	0,4	0,6	0,8	0,8	0,6	0,6	0,6
A3	0,4	0,6	0,6	0,6	0,8	0,8	0,6	0,6	0,6
A4	0,8	0,8	0,8	0,8	0,4	0,4	0,6	0,6	0,6

A5	0,8	0,8	0,2	0,8	0,6	0,6	0,4	0,4	0,6
Max	1	0,8	0,8	0,8	0,8	0,8	0,6	0,6	0,6

Nilai maksimum dalam setiap kriteria kemudian dibagi terhadap setiap nilai karyawan sesuai dengan kriteria masing-masing. Contoh pada karyawan A1 memiliki kriteria C1 dengan nilai 1 (sesuai tabel 3). Nilai 1 pada kriteria C1 dibagi terhadap nilai *Max* pada baris *Max*, kolom C1 pada tabel 3 yang memiliki nilai 1, hasil yang didapat adalah 1. Karyawan A2 yang memiliki nilai 0,6 pada kolom C1 dibagi terhadap nilai *Max* pada baris *Max*, kolom C1 yang bernilai 1. Hasil yang didapat adalah 0,6. Hasil proses kedua disajikan pada tabel 4 hasil pembagian:

Tabel 4 Pembagian nilai karyawan terhadap nilai maksimum kriteria

	C1	C2	C3	C4	C5	C6	C7	C8	C9
A1	1	1	1	1	1	1	1	1	1
A2	0,6	0,75	0,5	0,75	1	1	1	1	1
A3	0,4	0,75	0,75	0,75	1	1	1	1	1
A4	0,8	1	1	1	0,5	0,5	1	1	1
A5	0,8	1	0,25	1	0,75	0,75	0,666666667	0,666666667	1

Setiap kolom pada tabel tersebut kemudian dikali dengan nilai kriteria (pada tabel 1 dan 2) yang telah dibuat. Contoh pada baris A1 kolom C1 pada tabel 4 dikalikan dengan bobot kriteria C1, yaitu 0,05. Baris A1 kolom C2 pada tabel 4 dikalikan dengan bobot kriteria C2, dan seterusnya. Hasil nilai kriteria setiap karyawan dijumlahkan secara horizontal, dan hasil tersebut merupakan peringkat dari perhitungan dengan metode *simple additive weighting*. Berikut adalah tabel hasil perkalian serta penjumlahan nilai karyawan:

Tabel 5 Aspek Subjektif

	C1	C2	C3	C4	C5	C6	C7	C8	C9	Total
A1	0,05	0,05	0,05	0,05	0,1	0,25	0,15	0,15	0,15	1
A2	0,03	0,0375	0,025	0,0375	0,1	0,25	0,15	0,15	0,15	0,93
A3	0,02	0,0375	0,0375	0,0375	0,1	0,25	0,15	0,15	0,15	0,9325
A4	0,04	0,05	0,05	0,05	0,05	0,125	0,15	0,15	0,15	0,815
A5	0,04	0,05	0,0125	0,05	0,075	0,1875	0,1	0,1	0,15	0,765

Metode *simple additive weighting* merupakan metode yang memberikan nilai yang berbeda untuk setiap individu karyawan. Hasil perbandingan metode dengan yang dibuat oleh perusahaan menunjukkan metode ini membantu menemukan nilai yang berbeda ketika mendapatkan nilai kriteria yang sama antara karyawan satu dengan karyawan yang lain. Metode yang dibuat oleh perusahaan, tidak memiliki perbandingan nilai kriteria yang dapat menjadi ukuran perbedaan peringkat.

Contoh pada kasus ini sample dengan karyawan A2 dan A3 memiliki persamaan total nilai. Tabel 6 berikut merupakan data penilaian karyawan menggunakan metode yang digunakan oleh perusahaan. Hasil didapat adalah jumlah nilai A2 dan A3 seimbang.

Tabel 6 Nilai karyawan dengan metode perusahaan

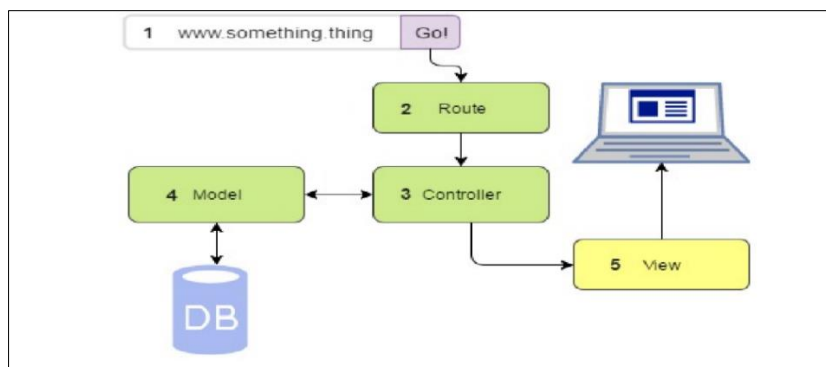
	C1	C2	C3	C4	C5	C6	C7	C8	C9	Jumlah
A2	0,0375	0,0375	0,025	0,0375	0,08	0,2	0,09	0,09	0,09	0,6875
A3	0,025	0,0375	0,0375	0,0375	0,08	0,2	0,09	0,09	0,09	0,6875

Sedangkan dengan menggunakan metode *simple additive weighting*, perusahaan bisa memilih kriteria mana yang dapat menjadi patokan pertama dalam penilaian karyawan. Sehingga, nilai dari kedua sample bisa berbeda meskipun total nilai sama. Ini disebabkan nilai setiap kriteria dihitung bersama tingkat prioritas kriteria (tabel 7).

Tabel 7 Nilai karyawan dengan metode *simple additive weighting*

	C1	C2	C3	C4	C5	C6	C7	C8	C9	Jumlah
A2	0,03	0,037 5	0,025	0,0375	0,1	0,25	0,15	0,15	0,15	0,93
A3	0,02	0,037 5	0,037 5	0,0375	0,1	0,25	0,15	0,15	0,15	0,9325

Framework Laravel dapat menjadi *framework* dalam penilaian karyawan ini dari sisi pemrograman karena kemudahannya dalam melakukan pemanggilan fungsi pada *view* hanya dengan menambahkan dua kali kurung kurawal “{}”. *Framework* ini juga dapat memanggil perintah *query builder* melalui *controller* dengan menggunakan *tools Eloquent* tanpa perlu mendeklarasikan fungsi kembali dalam kelas *Model*. *Framework Laravel* dalam penelitian ini menggunakan metode MVC yang secara umum terdiri dari empat proses utama yaitu *routing*, *controller*, *model*, dan *view* seperti pada gambar 5.

Gambar 5 Alur MVC pada *Laravel* (Budiman, Jamil, Hairah, Jati, & Rosmasari, 2018)

Routing dilakukan melalui *file web.php* dengan melakukan proses pemanggilan alamat *URL* dari berkas *view*. *URL* ini akan dipanggil menggunakan metode-metode dasar *HTTP* berupa *GET*, *POST*, *PUT*, serta *DELETE*. Proses *routing* ini akan dikirimkan kepada *controller* jika alamat *controller* diketahui. Proses berikut adalah proses yang dilakukan oleh *controller*. *Controller* akan menerima perintah dari proses *routing* tadi kemudian memasukkannya kedalam fungsi lain dapat berupa pengembalian nilai (*return*), dan menjalankan *model Eloquent* untuk mengambil data yang ada pada *database*. Proses pengembalian (*return*) juga dapat berupa pengembalian ke proses tampilan pengguna (proses *view*) atau pengembalian ke dalam bentuk notasi *javascript (JSON)*. Proses *model* dilakukan melalui proses pemanggilan *database* untuk mendapatkan semua data yang dipilih berdasarkan kueri. Contoh proses MVC pada penelitian ini adalah proses pengiriman nilai kriteria ke dalam *database*. Nilai kemudian dikirim menggunakan *ajax* dengan tipe pengiriman *PUT* setelah nilai dimasukkan semua oleh aktor

seperti pada contoh kode baris ke 3. *Route* yang dituju adalah rute dengan nama *updateNilaiKriteria* pada tabel baris kode 8 baris ke 4.

Tabel 8 Baris kode pengiriman *form* melalui *ajax*

1	<code>\$.ajax({</code>
2	<code> dataType: "json",</code>
3	<code> type: PUT,</code>
4	<code> url: {{route('updateNilaiKriteria')}},</code>
5	<code> data: form,</code>
6	<code> success: function (result) {</code>
7	<code> if (result == 'success') {</code>
8	<code> swal("Berhasil", "Nilai Telah Diperbaharui", "success")</code>
9	<code> }</code>
10	<code> console.log(result)</code>
11	<code> }</code>
12	<code>})</code>

Ajax melakukan proses pengiriman semua data tersebut dan memanggil perintah *route()* melalui file “*web.php*” yang berlokasi pada *folder* “*/route/web.php*”. Di dalam file “*web.php*” tersebut terdapat rute dengan nama *updateNilaiKriteria* yang merupakan nama lain untuk *class controller* “*KriteriaController*” dengan nama fungsi “*updateNilaiKriteria*” seperti pada contoh tabel baris kode 9.

Tabel 9 Baris kode *route*

1	<code>Route::put('/updateNilaiKriteria','KriteriaController@updateNilaiKriteria')</code>
2	<code>->name('updateNilaiKriteria');</code>

Dilakukan proses *request* nilai pada *form* yang telah dikirim oleh *ajax* pada tabel baris kode 10 fungsi *updateNilaiKriteria*. Proses *request* nilai menghasilkan sebuah *array* yang harus dipecah dengan bantuan fungsi *foreach* agar setiap nilai dapat disimpan ke dalam *database*. *Foreach* sendiri menghasilkan dua variabel yang berbeda nilai: “*\$rK*” dan “*\$val*” (baris 4). Variabel “*\$rK*” merupakan variabel yang berisi nomor *array*, sedangkan variabel “*\$val*” merupakan nilai yang dimasukkan oleh *user*. Di dalam *foreach* tersebut dibuat perintah untuk memanggil *model Eloquent* “*Kriteria*” dengan perintah mencari *id* dengan nilai variabel “*\$rK*”(baris 5). Variabel *karyawan* dibuat objek “*nilai*” dengan *input* berupa variabel “*\$val*” setelah mendapatkan data yang dimaksud. Proses penyimpanan nilai dilakukan dengan melakukan perintah “*save()*” (baris 7). Proses perulangan *foreach* dilakukan hingga perintah *save* berhenti dan dibuat sebuah variabel untuk mengirimkan perintah “*success*” kembali ke *ajax* melalui perintah “*response->json()*”.

Tabel 10 Baris kode *controller updateNilaiKriteria*

1	<code>public function updateNilaiKriteria(Request \$request)</code>
2	<code>{</code>
3	<code> \$idKr = \$request->id_kriteria;</code>
4	<code> foreach (\$reqNilai as \$rK => \$num) {</code>
5	<code> \$kriteria = Kriteria::find(\$rK);</code>
6	<code> \$kriteria->nilai_kriteria = \$num;</code>
7	<code> \$kriteria->save();</code>
8	<code> }</code>
9	<code> \$data = 'success';</code>
10	<code> return response()->json(\$data);</code>
11	<code>}</code>

5. Kesimpulan dan Saran

Kesimpulan dari penelitian ini adalah perancangan sistem penilaian karyawan dilakukan dengan cara membagi sistem dalam dua modul user beserta peran dan tugas yang dilakukan. Terdapat juga perhitungan metode *simple additive weighting* dalam sistem penilaian karyawan dilakukan dengan cara pemberian bobot kriteria, penilaian nilai kriteria setiap karyawan, mencari nilai tertinggi setiap kriteria, membagi nilai kriteria terhadap nilai tertinggi kriteria, proses perkalian dengan bobot kriteria, dan penjumlahan hasil nilai karyawan. Kesimpulan juga berakhir pada implementasi *simple additive weighting* dapat dilakukan dengan bantuan *library jQuery* untuk operasi matematis pada tabel dan alur program dasar *framework Laravel* yaitu mengatur *route*, memberikan fungsi pada *controller*, dan pengiriman data melalui model *eloquent*. Hasil penelitian ini juga mengarah pada kesimpulan lain yang menunjukkan bahwa menggunakan metode *simple additive weighting* dapat menunjukkan nilai antar karyawan yang berbeda bila dibandingkan dengan metode yang dipakai oleh perusahaan.

Ada beberapa saran yang dapat dilakukan dari hasil penelitian ini yang dapat dilakukan dalam pengembangan aplikasi dan implementasi untuk waktu yang akan datang:

1. Laravel merupakan framework dengan model MVC. Diperlukan pengembangan yang dilakukan dalam *framework* dengan bahasa pemrograman lain dan model MVC (seperti *django* yang berbasis *python*, *rails* yang berbasis *ruby* dan *sails.js* yang memiliki basis *javascript* agar dapat dibandingkan hasil dari setiap bahasa pemrograman dalam sisi performa, akurasi data, dan manajemen memori.
2. Implementasi sistem penilaian dengan menggunakan metode pendukung keputusan yang berbeda (sebagai contoh *AHP*, *TOPSIS*, dan *WP*) agar dapat dibandingkan performa dari setiap metode.
3. Implementasi sistem menggunakan teknologi *REST (Representational State Transfer)* agar proses penilaian dapat dilakukan melalui berbagai platform (ponsel pintar, dan komputer *desktop*).

Daftar Pustaka

- Asnawati, & Kanedi, I. (2012). Sistem Pendukung Keputusan Kenaikan Pangkat Karyawan Perseroan Terbatas Pelayaran Kumafa Lagun Marina Bengkulu. *Jurnal Media Infotama*, 8(1), 118–137. Retrieved from <http://jurnal.unived.ac.id/index.php/jmi/article/download/75/69>
- Budiman, E., Jamil, M., Hairah, U., Jati, H., & Rosmasari. (2018). Eloquent object relational mapping models for biodiversity information system. *Proceedings of the 2017 4th International Conference on Computer Applications and Information Processing Technology, CAIPT 2017, 2018-Janua(Caipt)*, 1–5. <https://doi.org/10.1109/CAIPT.2017.8320662>
- Chusna, N. A., Huboyo, H. S., & Andarani, P. (2017). Analisis Kebisingan Peralatan Pabrik Terhadap Daya Pendengaran Pekerja Di Pt . Pura Barutama Unit Pm 569 Kudus. *Teknik Lingkungan*, 6(1).
- Friyadie. (2016). Penerapan Metode Simpe Additive Weighting (SAW) Dalam Sistem Pendukung Keputusan Promosi Kenaikan Jabatan. *Jurnal Pilar Nusa Mandiri*, 11(1), 37–45.
- Hendrian, F., Devi, M., & Kurniawan, A. (2015). PENGEMBANGAN APLIKASI PENGUKURAN KINERJA KARYAWAN DENGAN MENGGUNAKAN FRAMEWORK LARAVEL. *Universitas Bina Nusantara*, 2.
- Karlitasari, L., Suhartini, D., & Benny. (2017). Comparison of simple additive weighting

- (SAW) and composite performance index (CPI) methods in employee remuneration determination. *IOP Conf. Series: Materials Science and Engineering* 166, 166(1). <https://doi.org/10.1088/1742-6596/755/1/011001>
- Masri, M. (2016). Penentuan Karyawan Terbaik Dengan Metode Simple Additive Weighting (PDAM Tirta Silaupiasa). *Journal of Electrical Technology*, 1(1), 36–41.
- Monteiro, F. (2018). *Hands-on full stack web development with Angular 6 and Laravel 5: Become fluent in both frontend and backend web development with Docker, Angular and Laravel*. Birmingham, UK: Packt Publishing.
- Nufus, H. (2016). Journal of Mathematics Education, Science and Technology. *Journal of Mathematics Education, Science and Technology*, 1(1), 125–136.
- Pura Group. (2018). *Tentang Kami*. Retrieved from Pura Group: <https://www.puragroup.com/id/tentang-kami/>
- Setiawan, N., Nasution, M. D. T. P., Rossanty, Y., Tambunan, A. R. S., Girsang, M., Agus, R. T. A., ... Nisa, K. (2018). Simple Additive Weighting as Decision Support System for Determining Employees Salary. *International Journal of Engineering & Technology*, 7(January), 309–313.
- Setiobudi, E. (2017). Analisis Sistem Penilaian Kinerja Karyawan. *Journal of Applied Business and Economics*, 3(3), 170–182. Retrieved from <http://journal.lppmunindra.ac.id/index.php/JABE/article/view/1768/1380>
- Shiddieq, D. F., & Septyan, E. (2017). Penilaian Kinerja Karyawan (Studi Kasus Di PT. Grafindo Media Pratama Bandung). *Lpkia*, 1(1), 1–7.
- Sinha, S. (2017). *Beginning laravel - A Beginners Guide to Application Development with Laravel 5.3*. Howrah, India: Apress.
- Stauffer, M. (2017). *Laravel: Up and Running: A Framework for Building Modern PHP Apps*. Sebastopol, CA: OReilly.
- Suprpty, B., Malani, R., & Nurhayati, O. D. (2016). Design of Information System for Acceptance Selection of Prospective Employees Online Using Tahani Fuzzy Logic Method and Simple Additive Weighting (SAW). *International Journal of Computing and Informatics (IJCANDI) Vol 1, No 1, February 2016, Pp. 17-28 ISSN:, 1(February)*, 17–28.
- Synta, F. E., Ginanjar, R., Juwono, R., Hadisukmana, N., & Wahyu, R. B. (2017). Online recruitment system for godrej indonesia. *Seminar Nasional APTIKOM (SEMNASTIKOM)*, (November), 244–249.
- Windu Gata, G. (2016). Pemodelan UML sistem informasi Monitoring Penjualan dan stok barang. *JURNAL KHATULISTIWA INFORMATIKA*, IV(2), 107–116. <https://doi.org/10.2135/cropsci1983.0011183X002300020002x>
- Yudhanto, Y., & Prasetyo, H. A. (2018). *Panduan Mudah Belajar Framework Laravel*. Jakarta: Elex Media Computindo.