

Implementasi Weighted Sum Model Pada Sistem Perekutan Karyawan di Agung's Collection

Dimas Dharu Ramadhan, *Retno Mumpuni, Andreas Nugroho Sihananto
Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan
Nasional Veteran Jawa Timur, Indonesia

Diterima: 20 April, 2025 | Revisi: 15 Mei, 2025 | Diterbitkan: 10 Juni 2025

DOI: <https://doi.org/10.33005/scan.v20i1.5660>

ABTSRAK

Perekutan karyawan yang tepat merupakan salah satu faktor kunci dalam menunjang keberhasilan perusahaan. Agung's Collection, sebuah perusahaan di industri tekstil, menghadapi tantangan dalam menyeleksi calon karyawan yang memenuhi kualifikasi yang diinginkan. Oleh karena itu, penelitian ini bertujuan untuk menerapkan metode Weighted Sum Model (WSM) dalam proses perekutan karyawan guna membantu pengambilan keputusan yang lebih objektif. WSM digunakan karena kemampuannya untuk mengevaluasi calon berdasarkan berbagai kriteria. Setiap kriteria diberikan bobot sesuai dengan tingkat kepentingannya dalam peran yang dibutuhkan. Data karyawan diolah menggunakan metode ini, dan hasilnya berupa peringkat dari setiap kandidat yang membantu perusahaan memilih kandidat terbaik. Implementasi WSM terbukti efektif dalam mempermudah dan meningkatkan akurasi proses perekutan, serta mengurangi subjektivitas dalam penilaian. Dengan metode ini, Agung's Collection diharapkan dapat memperoleh karyawan yang berkualitas dan sesuai dengan kebutuhan perusahaan.

Kata Kunci: Weighted Sum Model, perekutan karyawan, penilaian kriteria, pengambilan keputusan

Implementation of the Weighted Sum Model in Employee Recruitment at Agung's Collection

ABSTRACT

The proper recruitment of employees is one of the key factors in supporting a company's success. Agung's Collection, a company in the textile industry, faces challenges in selecting candidates who meet the desired qualifications. Therefore, this study aims to apply the Weighted Sum Model (WSM) in the employee recruitment process to assist in making more objective decisions. WSM is used for its ability to evaluate candidates based on various criteria. Each criterion is weighted according to its importance in the required role. Employee data is processed using this method, resulting in a ranking of candidates, which helps the company select the best candidate. The implementation of WSM has proven effective in simplifying and improving the accuracy of the recruitment process, as well as reducing subjectivity in evaluations. With this method, Agung's Collection is expected to acquire qualified employees that meet the company's needs.

Keywords: Weighted Sum Model, employee recruitment, criteria evaluation, decision making

*Corresponding Author:

Email : retnomumpuni.if@upnjatim.ac.id
Alamat : Jl. Rungkut Madya, Gn. Anyar, Kec. Gn. Anyar, Surabaya, Jawa Timur 60294

This article is published under a Creative Commons Attribution 4.0 International License.



PENDAHULUAN

Perekutan karyawan yang tepat merupakan salah satu faktor penting yang menentukan keberhasilan sebuah perusahaan, terutama dalam industri yang sangat kompetitif seperti tekstil. Agung's Collection, sebuah perusahaan yang bergerak di bidang tekstil, selama ini menggunakan metode manual dalam proses perekutan karyawannya. Sistem manual ini cenderung kurang efektif dan berisiko menimbulkan bias dalam pengambilan keputusan, karena bergantung pada penilaian subjektif dari pihak perekut. Selain itu, proses seleksi manual sering kali membutuhkan waktu yang lebih lama dan tidak memiliki sistem evaluasi yang terukur secara konsisten.

Dalam menghadapi tantangan tersebut, Agung's Collection perlu menerapkan metode yang lebih objektif dan efisien untuk menyeleksi calon karyawan berdasarkan kualifikasi yang relevan dengan kebutuhan perusahaan. Salah satu metode yang dapat digunakan adalah Weighted Sum Model (WSM), yang merupakan salah satu teknik dalam pengambilan keputusan multi-kriteria. WSM memungkinkan evaluasi terhadap sejumlah kriteria yang telah ditetapkan sebelumnya, seperti pengalaman kerja, kemampuan teknis, dan keterampilan komunikasi. Setiap kriteria diberikan bobot sesuai dengan tingkat kepentingannya, sehingga proses seleksi menjadi lebih terstruktur dan obyektif.

Penelitian ini akan fokus pada implementasi WSM di Agung's Collection untuk membantu perusahaan dalam memilih karyawan yang paling sesuai dengan kebutuhan mereka. Diharapkan dengan penerapan metode ini, Agung's Collection dapat meningkatkan kualitas dan akurasi dalam proses perekutan, mengurangi bias penilaian, serta memperoleh karyawan yang mampu mendukung perkembangan perusahaan ke arah yang lebih baik.

METODE PENELITIAN

Spesifikasi Perangkat Keras

Tabel 1 diatas adalah spesifikasi perangkat keras yang peneliti gunakan pada penelitian kali ini.

Spesifikasi Perangkat Lunak

Tabel 2 adalah spesifikasi perangkat lunak yang peneliti gunakan pada penelitian kali ini.

Perancangan Sistem

Dalam merancang sebuah sistem kita perlu merancang alur kerja sistem terlebih dahulu seperti perancangan Flowchart, Use Case Diagram, Activity Diagram, Sequence Diagram dan juga merancang basis data melalui ERD (Entity Relationship Diagram).

Use Case Diagram

Diagram use case pada Gambar 1 menggambarkan interaksi antara Admin dan Pelamar dengan sistem untuk proses rekrutmen berbasis Weighted Sum Model (WSM). Admin memiliki beberapa tugas utama, termasuk mengatur kriteria WSM, melihat daftar pelamar, mengatur waktu wawancara, memberikan nilai WSM, serta menerima atau menolak pelamar. Semua tugas ini bergantung pada tindakan LOGIN sebagai syarat awal untuk mengakses fungsionalitas sistem. Di sisi lain, Pelamar hanya memiliki satu use case utama, yaitu Mengajukan Lamaran, yang juga memerlukan login sebagai

prasyarat. Setiap use case yang dijalankan oleh Admin dan Pelamar menyertakan include ke login, artinya login menjadi bagian penting sebelum melakukan interaksi lebih lanjut dengan sistem.

Activity Diagram

Activity Diagram Kriteria

Pada Gambar 2 menjelaskan alur bagaimana pembuatan kriteria baru. Yang pertama dilakukan adalah memasuki menu kriteria lalu menekan tombol tambah kriteria dan selanjutnya akan muncul halaman berisi form data kriteria yang harus diisi, setelah terisi semua dan menekan tombol submit maka sistem akan menerima inputan tersebut dan melakukan cek apakah semua data sudah terisi dan juga melakukan cek apakah nama kriteria yang digunakan sudah ada sebelumnya atau tidak. Jika kedua kondisi di atas tidak terpenuhi maka kriteria baru tidak dapat dibuat dan akan muncul pesan galat, namun jika kedua kondisi di atas terpenuhi maka data kriteria bisa disimpan pada basis data.

Tabel 1
Spesifikasi Perangkat Keras

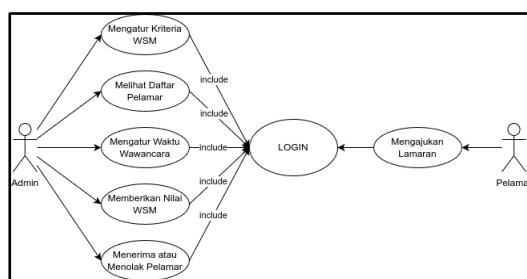
Merk	Acer Aspire z3-451
Processor	AMD A-Series quad-core processor A10-5757M (4 MB L2 cache, up to 3.5 GHz, 35W)
Graphic	AMD Radeon HD8650G
Memory	4GB DDR3L SDRAM
Storage	500GB HDD

Sumber: Data Diolah

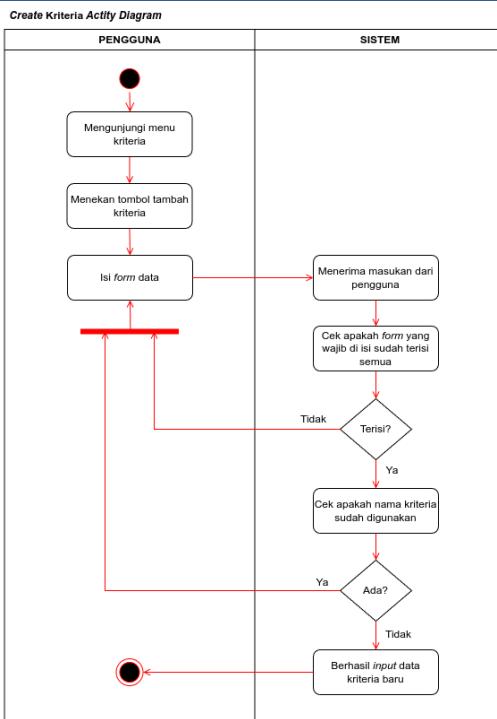
Tabel 2
Spesifikasi Perangkat Lunak

Nama	Tipe
Sistem Operasi	Ubuntu 20.04.6 LTS
VS Code	Versi 1.88.1
XAMPP	Versi 8.2.4-0
PHP	Versi 8.2.15
Laravel	Versi 10.44.0
MySQL	8.0.36-Ubuntu0.20.04.1 for Linux on x86_64

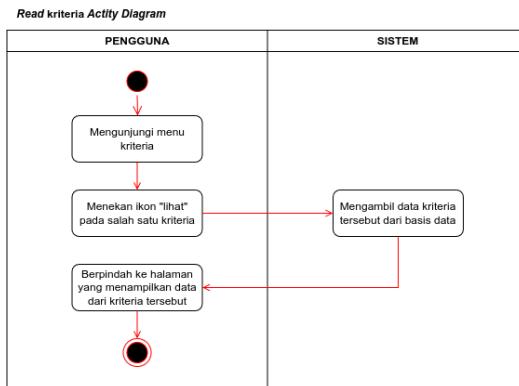
Sumber: Data Diolah



Gambar 1. Use Case Diagram



Gambar 2. Activity Diagram Create Kriteria



Gambar 3. Activity Diagram Read Kriteria

Pada Gambar 3 menunjukkan alur dari read/melihat data kriteria, langkah pertama yang dilakukan adalah masuk ke menu kriteria lalu menekan ikon “lihat” pada salah satu data yang ingin dilihat, setelah itu sistem akan mengambil data tersebut dari database dan halaman langsung berpindah ke halaman yang memperlihatkan data dari kriteria tersebut.

Gambar 4 memperlihatkan alur dari proses update kriteria. Proses dimulai dengan mengunjungi menu kriteria lalu menekan ikon “update” pada salah satu kriteria lalu akan muncul data dari kriteria tersebut yang bisa diubah, setelah data diubah dan di submit maka inputan tersebut akan dicek apakah data kriterianya sudah terisi semua atau belum, dan melakukan cek pada nama kriteria apakah sudah ada nama kriteria yang sama pada basis data atau tidak, jika ada kondisi yang tidak terpenuhi dari 2 syarat di atas maka data tidak dapat dikirim ke basis data dan akan menampilkan pesan galat,

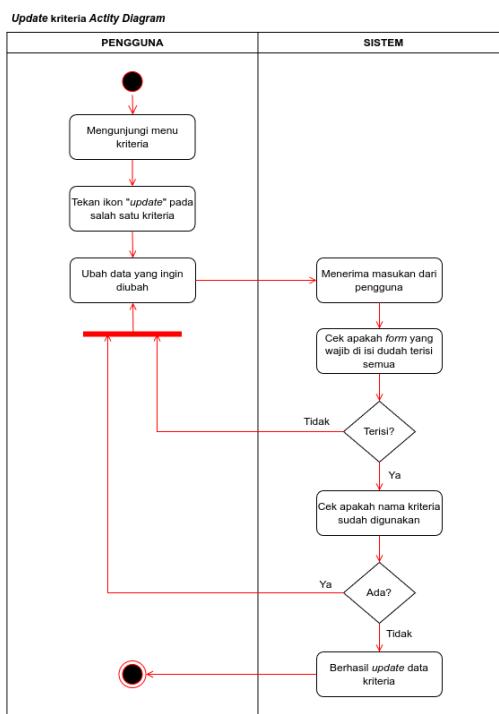
namun jika 2 syarat tadi terpenuhi maka data yang sudah diperbarui akan terkirim ke basis data.

Gambar 5 memperlihatkan alur dari hapus kriteria, langkah pertama adalah masuk ke halaman kriteria lalu menekan ikon “hapus” pada salah satu data kriteria yang ingin dihapus, setelah itu sistem akan memunculkan alert konfirmasi apakah yakin ingin menghapus kriteria, jika menekan tombol “yakin” maka kriteria akan terhapus tapi jika menekan “batal” maka penghapusan kriteria dibatalkan.

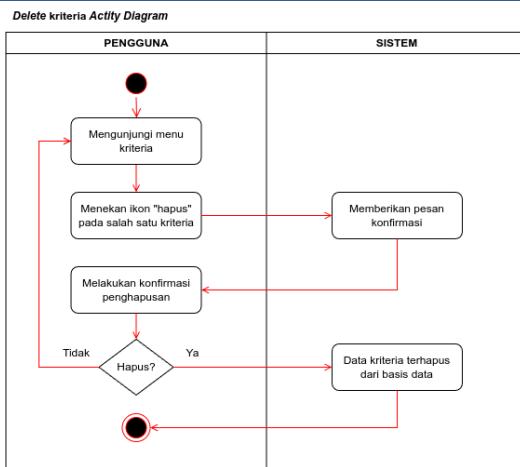
Activity Diagram Pelamar

Pada Gambar 6 menjelaskan alur bagaimana pembuatan pelamar baru. Yang pertama dilakukan adalah memasuki menu lowongan pekerjaan lalu akan muncul halaman berisi form data pelamar yang harus diisi, setelah terisi semua dan menekan tombol submit maka sistem akan menerima inputan tersebut dan melakukan cek apakah semua data sudah terisi. Jika kondisi di atas tidak terpenuhi maka pelamar baru tidak dapat dibuat dan akan muncul pesan galat, namun jika kondisi di atas terpenuhi maka data pelamar bisa disimpan pada basis data. Sesaat setelah berhasil mengajukan lamaran, maka sistem juga akan mengirimkan pesan whatsapp otomatis ke admin yang memberitahukan bahwa ada pelamar baru yang mendaftar.

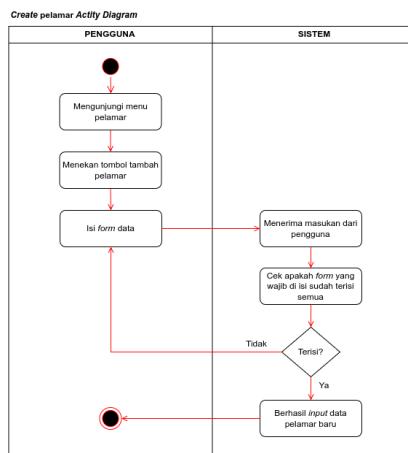
Pada Gambar 7 memperlihatkan alur dari read/melihat data pelamar, langkah pertama yang dilakukan adalah masuk ke menu pelamar lalu menekan ikon “lihat” pada salah satu data yang ingin dilihat, setelah itu sistem akan mengambil data tersebut dari database dan halaman langsung berpindah ke halaman yang memperlihatkan data dari pelamar tersebut.



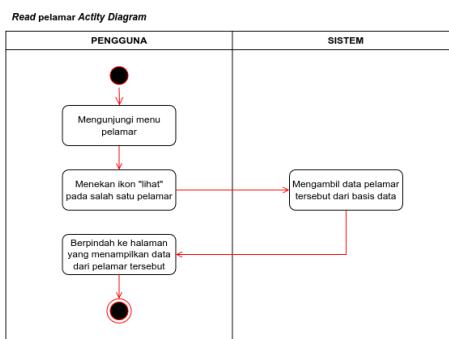
Gambar 4. Activity Diagram Update Kriteria



Gambar 5. Activity Diagram Delete Kriteria

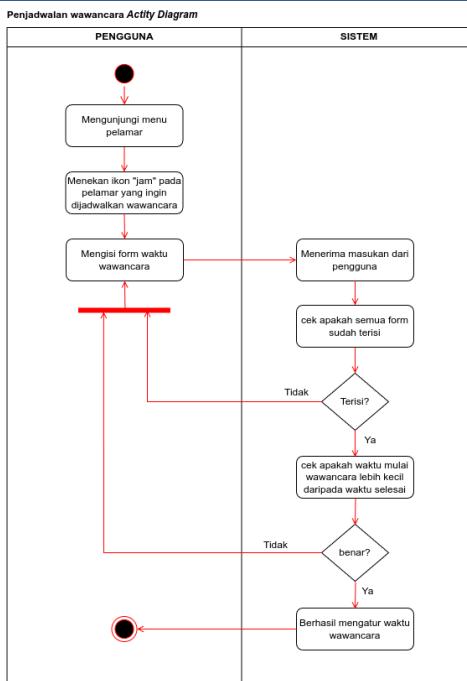


Gambar 6. Activity Diagram Create Pelamar

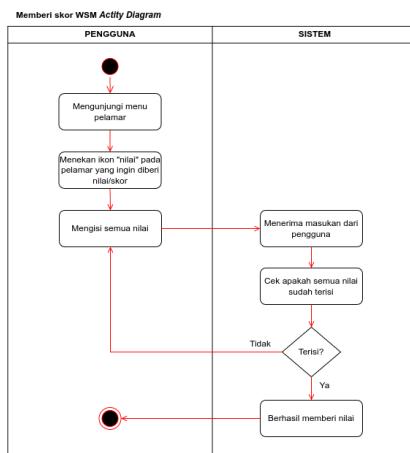


Gambar 7. Activity Diagram Read Pelamar

Gambar 8 menunjukkan activity diagram dari penjadwalan wawancara, langkah pertama yang dilakukan adalah menekan ikon “jam” pada pelamar yang ingin diatur waktu wawancaranya, lalu setelah itu akan tampil form dan isikan waktu mulai wawancara dan selesai wawancara, jika semua form sudah terisi semua maka penjadwalan wawancara telah berhasil. Pada penjadwalan wawancara juga kami buat agar mengirimkan jam wawancara otomatis ke whatsapp pelamar setelah memasukkan waktu wawancara, dan juga otomatis membuat jadwal pada google kalender admin.



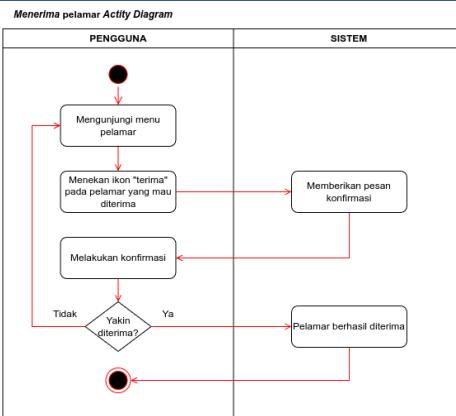
Gambar 8. Activity Diagram Penjadwalan Wawancara



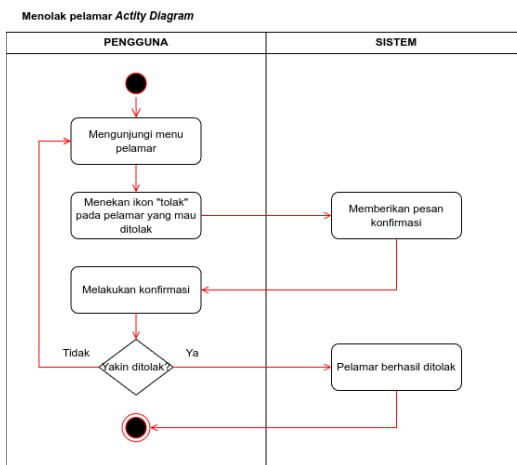
Gambar 9. Activity Diagram Memberi Skor WSM

Gambar 9 menunjukkan activity diagram dalam pemberian skor WSM (Weighted Sum Model). Tahap ini dimulai dengan mengunjungi menu pelamar dan menekan ikon “nilai” pada pelamar yang ingin diberi nilai, setelah itu akan muncul beberapa variabel nilai yang berasal dari database kriteria dan silakan isi semua nilai pada kriteria-kriteria tersebut, setelah terisi semua maka nilai WSM akan dihitung dan tersimpan di basis data dan dapat digunakan untuk membantu mengambil keputusan apakah akan menerima atau menolak pelamar tersebut.

Gambar 10 memperlihatkan alur dari menerima pelamar, langkah pertama adalah masuk ke halaman pelamar lalu menekan ikon “terima” pada salah satu pelamar yang ingin diterima, setelah itu sistem akan memunculkan alert konfirmasi apakah yakin ingin menerima pelamar tersebut, jika menekan tombol “yakin” maka pelamar akan diterima dan statusnya berubah menjadi “Diterima” dan akan mengirimkan pesan whatsapp otomatis ke pelamar bahwa dia dinyatakan diterima.



Gambar 10. Activity Diagram Menerima Pelamar



Gambar 11. Activity Diagram Menolak Pelamar

Gambar 11 memperlihatkan alur dari menolak pelamar, langkah pertama adalah masuk ke halaman pelamar lalu menekan ikon "tolak" pada salah satu pelamar yang ingin ditolak, setelah itu sistem akan memunculkan alert konfirmasi apakah yakin ingin menolak pelamar tersebut, jika menekan tombol "yakin" maka pelamar akan ditolak dan statusnya berubah menjadi "Ditolak" dan akan mengirimkan pesan whatsapp otomatis ke pelamar bahwa dia dinyatakan ditolak.

Sequence Diagram

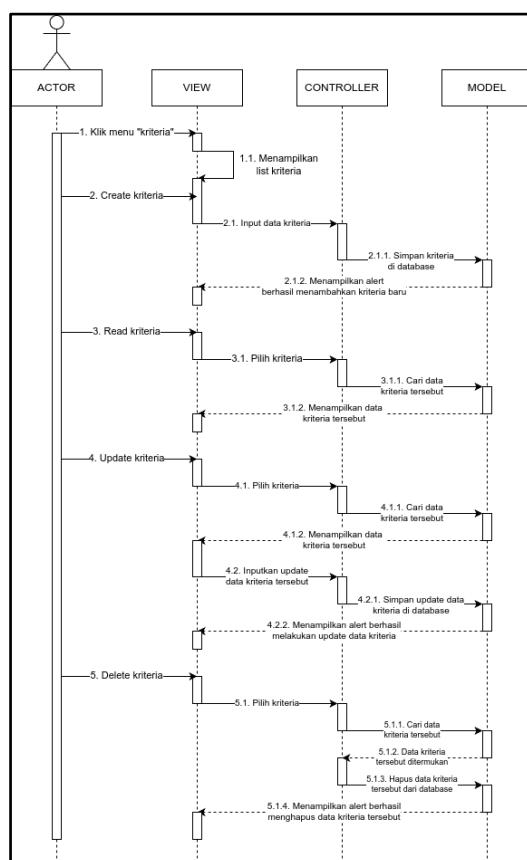
Sequence Diagram Kriteria

Gambar 12 menunjukkan alur pengelolaan data kriteria dalam sistem melalui proses CRUD (Create, Read, Update, Delete). Proses dimulai ketika pengguna memilih menu "kriteria," yang membuat sistem menampilkan daftar kriteria yang ada. Untuk menambah kriteria baru, pengguna memasukkan data melalui tampilan, dan data tersebut kemudian disimpan oleh controller ke dalam database. Jika pengguna ingin membaca data kriteria, sistem akan mengambil data dari database dan menampilkannya. Untuk memperbarui kriteria, pengguna memilih kriteria yang ingin diubah, memasukkan data baru, dan controller akan memperbarui informasi tersebut di database. Dalam kasus penghapusan, pengguna memilih kriteria yang ingin dihapus, dan controller menghapus data tersebut dari database, disertai dengan notifikasi keberhasilan. Diagram ini

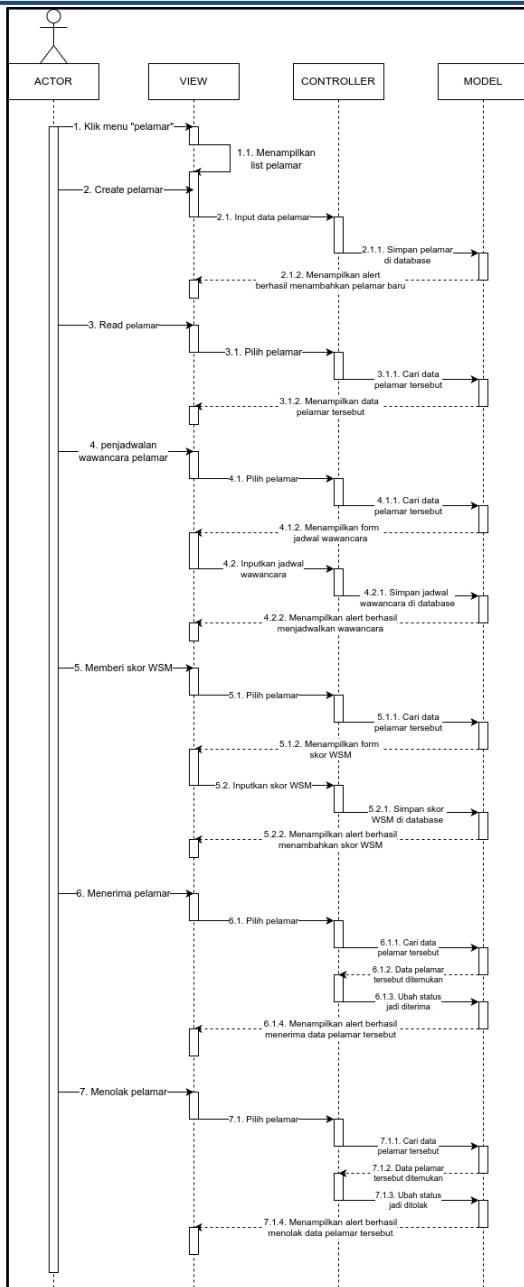
menggambarkan interaksi antara pengguna, tampilan, controller, dan model dalam mengelola data kriteria.

Sequence Diagram Pelamar

Diagram pada Gambar 13 menggambarkan proses CRUD (Create, Read, Update, Delete) untuk mengelola data pelamar dalam suatu sistem. Proses dimulai dari aktor yang mengklik menu "pelamar" untuk melihat daftar pelamar. Selanjutnya, aktor dapat menambahkan data pelamar baru, membaca data pelamar yang sudah ada, menjadwalkan wawancara, memberikan skor WSM, serta menerima atau menolak pelamar. Setiap aksi yang dilakukan aktor melibatkan tampilan (view), pengendali (controller), dan model data untuk memproses dan menyimpan data pelamar serta memberikan notifikasi keberhasilan aksi.



Gambar 12. Sequence Diagram Kriteria



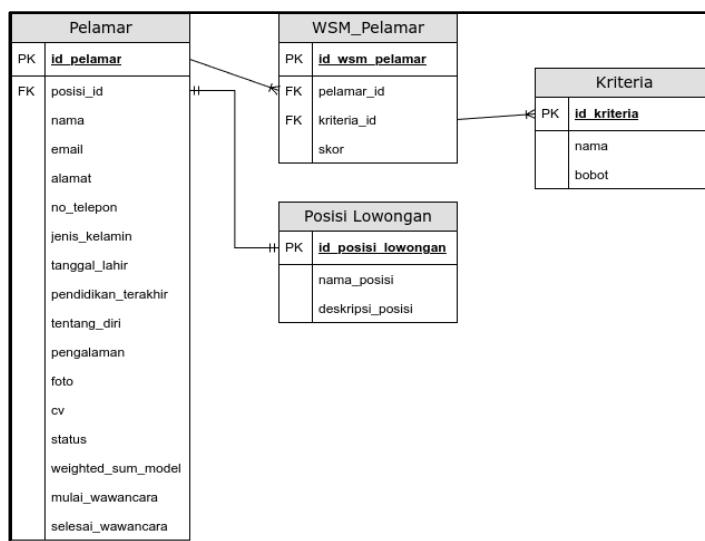
Gambar 13. Sequence Diagram Pelamar

Entity Relationship Diagram

Gambar 14 merupakan ERD yang menggambarkan hubungan antara beberapa entitas dalam sistem rekrutmen. Pelamar adalah entitas utama yang menyimpan data pelamar seperti nama, alamat, pendidikan, pengalaman, serta informasi wawancara. Setiap pelamar memiliki relasi foreign key ke entitas Posisi Lowongan, yang menyimpan detail posisi pekerjaan yang dilamar seperti nama dan deskripsi posisi. Selain itu, terdapat atribut weighted_sum_model yang mungkin menyimpan hasil skor WSM dari pelamar.

Entitas WSM_Pelamar berfungsi untuk mencatat skor WSM berdasarkan kriteria tertentu yang dihubungkan dengan Kriteria, yang memiliki atribut seperti nama dan bobot. Relasi ini mengindikasikan bahwa setiap pelamar dievaluasi berdasarkan

beberapa kriteria, dengan skor yang disimpan dalam entitas WSM_Pelamar. Relasi antar tabel memastikan sistem dapat melacak informasi pelamar, posisi lowongan, dan kriteria penilaian.



Gambar 14. Entity Relationship Diagram

HASIL DAN PEMBAHASAN

Implementasi WSM

Implementasi algoritma Weighted Sum Model (WSM) digunakan dalam fitur pelamar yang nantinya digunakan untuk membantu mengambil keputusan pelamar mana yang harus diterima berdasarkan pemeringkatan skor yang diberikan. Variabel yang diperlukan untuk perhitungan WSM terdiri dari kriteria, bobot, dan skor. Kode di bawah merupakan proses perhitungan WSM yang kami terapkan pada aplikasi kami.

Script Perhitungan WSM	
1	protected function calculateWSM(array \$validated): float
2	{
3	\$wsmValue = 0;
4	
5	\$kriteria = Kriteria::all();
6	
7	foreach (\$kriteria as \$kriterium) {
8	\$columnName = strtolower(str_replace(' ', '_', \$kriterium->nama));
9	\$skor = \$validated[\$columnName];
10	\$bobot = \$kriterium->bobot;
11	
12	\$wsmValue += \$skor * \$bobot;
13	}
14	
15	return \$wsmValue;
16	}

Metode “calculateWSM” pada kode di atas digunakan untuk menghitung nilai Weighted Scoring Model (WSM) dari seorang pelamar berdasarkan kriteria yang telah divalidasi. Fungsi ini mengiterasi setiap kriteria yang ada di dalam basis data, lalu mengambil skor yang relevan dari input pelamar. Setiap skor dikalikan dengan bobot kriteria yang bersangkutan, dan hasilnya ditambahkan ke total nilai WSM.

Script menyimpan dan memperbarui hasil WSM	
1 protected function saveWsmPelamar(Pelamar \$pelamar, array \$validated): void 2 { 3 \$krriteria = Kriteria::all(); 4 foreach (\$krriteria as \$kriterium) { 5 \$columnName = strtolower(str_replace(' ', '_', \$kriterium->nama)); 6 \$skor = \$validated[\$columnName]; 7 8 \$wsmPelamar = WsmPelamar::updateOrCreate(9 ['pelamar_id' => \$pelamar->id, 10 'kriteria_id' => \$kriterium->id], ['skor' => \$skor] 11); 12 } 13 }	

Metode “saveWsmPelamar” pada kode di atas bertujuan untuk menyimpan atau memperbarui skor WSM dari seorang pelamar ke dalam tabel WsmPelamar. Fungsi ini juga mengiterasi setiap kriteria yang ada dan mengambil skor yang sesuai dari input pelamar. Kemudian, fungsi ini menggunakan metode updateOrCreate dari Eloquent untuk mencari apakah sudah ada entri di tabel WsmPelamar untuk kombinasi pelamar_id dan kriteria_id yang spesifik. Jika sudah ada, entri tersebut akan diperbarui dengan skor baru; jika belum ada, entri baru akan dibuat. Dengan cara ini, fungsi ini memastikan bahwa setiap skor kriteria untuk pelamar disimpan atau diperbarui dengan benar di basis data.

Hasil Tampilan Menu Kriteria

Gambar 15 merupakan tampilan dari daftar kriteria. Terdapat tabel berisi data kriteria yang sudah ada pada database. Gambar 16 merupakan tampilan dari create kriteria yang terdapat banyak form yang dapat diisi untuk membuat kriteria baru. Gambar 17 merupakan tampilan ketika melakukan update kriteria, untuk tampilannya sama saja dengan saat create, terdapat banyak form yang sudah terisi data sebelumnya dan tinggal diganti saja. Di Bagian pojok kanan bawah terdapat banner kecil berwarna merah yang bertuliskan “Saved successfully” ketika berhasil melakukan update data.

No	Nama Kriteria	Bobot	Action
1	Keterampilan Managerial	0.10	[Edit] [Delete] [Create]
2	Ambisi	0.05	[Edit] [Delete] [Create]
3	Kepatuhan Standar	0.05	[Edit] [Delete] [Create]
4	Problem Solving	0.15	[Edit] [Delete] [Create]
5	Kemampuan Komunikasi	0.15	[Edit] [Delete] [Create]
6	Kreativitas	0.10	[Edit] [Delete] [Create]
7	Pengalaman Teknis	0.20	[Edit] [Delete] [Create]
8	Pengalaman Kerja	0.20	[Edit] [Delete] [Create]

Gambar 15. Tampilan Daftar Kriteria

Create Kriteria

Nama Kriteria *	Bobot (Total bobot semua kriteria harus 1 agar optimal) *
<input type="text" value="asdasd"/>	<input type="text" value="0.13"/>
Back to Index Create	

Gambar 16. Tampilan Create Kriteria

Edit Kriteria

Nama Kriteria *	Bobot (Total bobot semua kriteria harus 1 agar optimal) *
<input type="text" value="asdasd"/>	<input type="text" value="0.13"/>
Back to Index Create Update	

Save successfully.

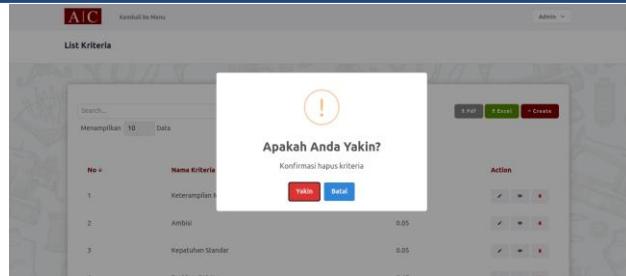
Gambar 17. Tampilan Update Kriteria

Show Kriteria

Nama Kriteria	Created At
Keterampilan Managerial	2024-08-12 14:05:13
Bobot	Updated At
0.10	2024-08-12 14:05:13
Back to Index Create	

Gambar 18. Tampilan Show Kriteria

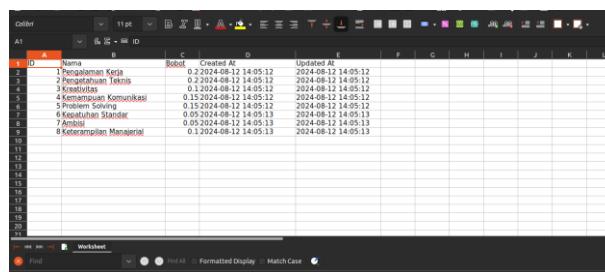
Gambar 18 merupakan tampilan show kriteria yang berfungsi untuk melihat keseluruhan data pada kriteria yang dipilih. Gambar 19 merupakan tampilan ketika akan melakukan delete kriteria, terdapat alert untuk menyakinkan apakah ingin menghapus data ini dan menghindari terjadinya tidak sengaja tertekan. Gambar 20 merupakan tampilan dari hasil export pdf, hasilnya berupa tabel yang didalamnya terdapat semua data dari kriteria yang sudah terdaftar pada database kriteria.



Gambar 19. Tampilan Delete Kriteria



Gambar 20. Tampilan Export PDF Kriteria



Gambar 21. Tampilan Export Excel Kriteria

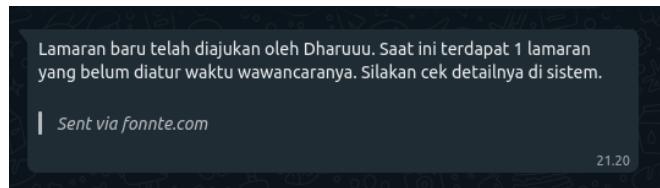
Gambar 22. Tampilan Ajukan Lamaran

Gambar 21 merupakan tampilan dari hasil export excel, hasilnya berupa data-data dari database kriteria yang di tampilkan di dalam excel seperti pada gambar di atas.

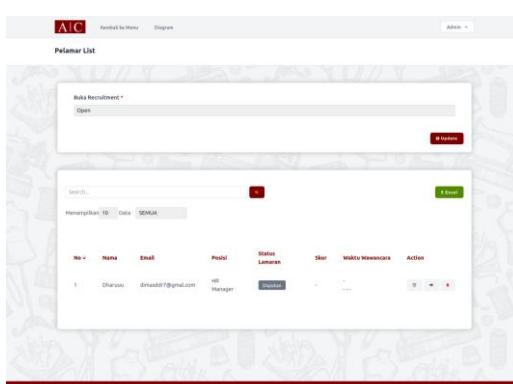
Hasil Tampilan Menu Pelamar

Gambar 22 merupakan tampilan ketika akan mengajukan lamaran, terdapat form-form data diri yang harus diisi agar bisa mengajukan lamaran baru. Gambar 23 merupakan pesan notifikasi yang didapatkan admin ketika ada pelamar baru yang masuk kedalam

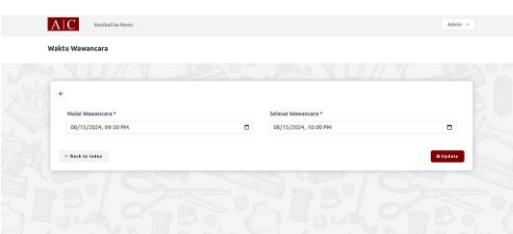
database. Gambar 24 merupakan tampilan dari daftar pelamar, terdapat pilihan untuk membuka atau menutup lamaran pada bagian atas dan bagian bawah merupakan daftar dari pelamar. Gambar 25 diatas merupakan tampilan ketika menentukan waktu wawancara, terdapat 2 inputan yaitu waktu mulai dan waktu selesai wawancara. Setelah memasukkan waktu wawancara maka pelamar akan mendapatkan pesan yang berisi waktu wawancara dan detail lainnya seperti pada gambar 26.



Gambar 23. Pesan Notifikasi Lamaran Baru



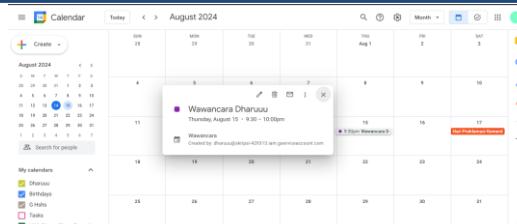
Gambar 24. Tampilan Daftar Pelamar



Gambar 25. Tampilan Menentukan Waktu Wawancara



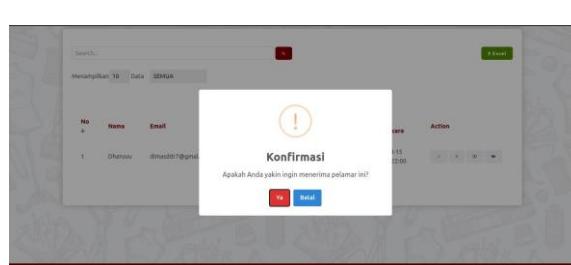
Gambar 26. Pesan Pemberitahuan Waktu Wawancara



Gambar 27. Jadwal Wawancara pada Google Calendar

Gambar 28. Tampilan Memberi Nilai WSM Pelamar

Gambar 29. Tampilan Show Pelamar



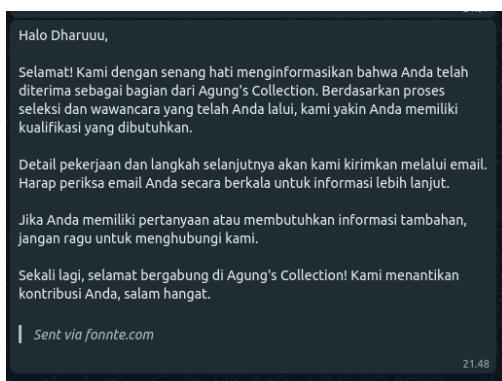
Gambar 30. Tampilan Alert Terima Pelamar

Setelah memasukkan waktu wawancara, admin juga akan mendapatkan pengingat di google calendar seperti yang terlihat pada gambar 27. Pada gambar 28 terlihat tampilan ketika akan memberikan nilai pelamar, terdapat beberapa inputan yang berasal dari kriteria yang sudah ditentukan, setelah diisi maka nilai WSM akan dihitung oleh sistem. Pada gambar 29 terlihat tampilan show pelamar yang terdapat data pribadi pelamar, waktu wawancara, dan nilai wawancara yang telah diberikan serta hasil perhitungan

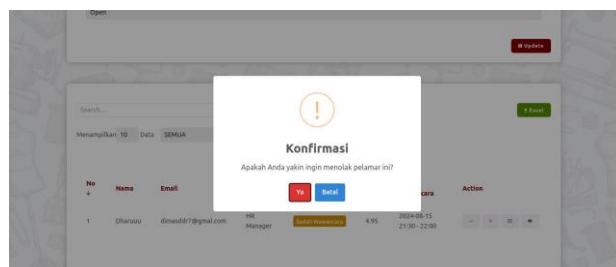
WSM. Pada Gambar 30 di atas terlihat alert ketika ingin menerima pelamar dan ketika menekan yakin maka pelamar akan menerima pesan bahwa dia diterima seperti pada Gambar 31. Pada Gambar 32 di atas terlihat alert ketika ingin menolak pelamar dan ketika menekan yakin maka pelamar akan menerima pesan bahwa dia ditolak seperti pada Gambar 33.

Hasil Pengujian Black Box

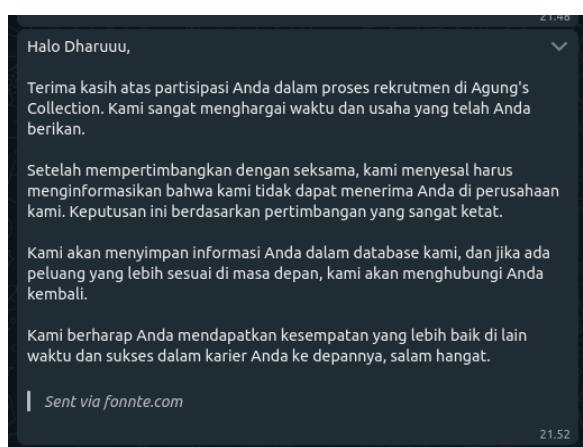
Pengujian black box bertujuan untuk memastikan semua fitur di dalam aplikasi yang telah dibuat berjalan sesuai dengan yang diharapkan, pengujian ini dilakukan secara manual dengan mencoba satu persatu fungsionalitas dari setiap fitur, untuk detail dari hasil pengujian black box pada tiap-tiap fitur akan ditampilkan pada tabel-tabel di bawah.



Gambar 31. Pesan Bahwa Pelamar Telah Diterima



Gambar 32. Jadwal Wawancara pada Google Calendar



Gambar 33. Jadwal Wawancara pada Google Calendar

Tabel 3
Hasil Pengujian Black Box Kriteria

Nama Pengujian	Hasil Pengujian	Kesimpulan
Mengunjungi menu kriteria	Berhasil mengunjungi menu kriteria	Sesuai
Menambahkan data kriteria baru	Data kriteria baru berhasil ditambahkan	Sesuai
Menampilkan data kriteria	Data kriteria berhasil ditampilkan	Sesuai
Memperbarui data kriteria	Data kriteria berhasil diperbarui	Sesuai
Hapus data kriteria	Data kriteria berhasil dihapus	Sesuai
Melakukan export PDF data kriteria	Berhasil melakukan export PDF data kriteria	Sesuai
Melakukan export excel data kriteria	Berhasil melakukan export excel data kriteria	Sesuai

Sumber: Data Diolah

Tabel 4
Hasil Pengujian Black Box Pelamar

Nama Pengujian	Hasil Pengujian	Kesimpulan
Mengunjungi menu pelamar	Berhasil mengunjungi menu pelamar	Sesuai
Menambahkan data pelamar baru	Data pelamar baru berhasil ditambahkan	Sesuai
Menampilkan data pelamar	Data pelamar berhasil ditampilkan	Sesuai
Menentukan waktu wawancara pelamar	Berhasil menambahkan waktu wawancara	Sesuai
Memberi nilai pelamar	Berhasil memberikan nilai pelamar	Sesuai
Menerima pelamar	Berhasil menerima pelamar	Sesuai
Menolak pelamar	Berhasil menolak pelamar	Sesuai

Sumber: Data Diolah

Tabel 3 merupakan hasil pengujian fitur kriteria, terdapat 7 pengujian dan untuk semua hasil pengujinya sesuai dengan yang diharapkan. Tabel 4 merupakan hasil pengujian fitur pelamar, terdapat 7 pengujian dan untuk semua hasil pengujinya sesuai dengan yang diharapkan.

SIMPULAN

Penelitian ini berhasil mengimplementasikan metode Weighted Sum Model (WSM) dalam proses perekrutan karyawan di Agung's Collection. Hasil dari penerapan WSM menunjukkan bahwa metode ini dapat membantu meningkatkan akurasi dan objektivitas dalam menyeleksi calon karyawan. Dengan memberikan bobot pada setiap kriteria sesuai dengan tingkat kepentingannya, perusahaan dapat mengurangi subjektivitas dalam proses pengambilan keputusan. WSM tidak hanya mempermudah perusahaan dalam menentukan kandidat terbaik, tetapi juga memungkinkan proses perekrutan yang lebih efisien dan terstruktur. Melalui metode ini, Agung's Collection mampu mengidentifikasi karyawan yang berkualitas dan sesuai dengan kebutuhan spesifik

perusahaan, yang pada akhirnya dapat meningkatkan kinerja dan keberhasilan perusahaan dalam jangka panjang.

DAFTAR PUSTAKA

- [1] E. Elgamar, BUKU AJAR KONSEP DASAR PEMROGRAMAN WEBSITE DENGAN PHP. Ahlimedia Book, 2020.
- [2] A. Jimi, "Rancang Bangun sistem informasi desa berbasis website (Studi Kasus Desa Netpala)," *Jurnal Pendidikan Teknologi Informasi (JUKANTI)*, vol. 2, no. 1, pp. 1–7, 2019. doi: 10.37792/jukanti.v2i1.17.
- [3] A. P. U. Siahaan, "Tuition reduction determination using fuzzy tsukamoto," Center for Open Science, 2017. doi: 10.31227/osf.io/94rdz.
- [4] D. Purnama Sari and R. Wijanarko, "Implementasi Framework Laravel pada Sistem Informasi Penyewaan Kamera (Studi Kasus di Rumah Kamera Semarang)," *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, vol. 2, no. 1, p. 32, 2020. doi: 10.36499/jinrpl.v2i1.3190.
- [5] A. R. Mulyanto, *Rekayasa Perangkat Lunak Jilid 1*. Jakarta: Direktorat Pembinaan Sekolah Menengah Kejuruan, Direktorat Jenderal Manajemen Pendidikan Dasar dan Menengah, Departemen Pendidikan Nasional, 2008.
- [6] M. Tabrani and I. R. Aghniya, "Implementasi metode waterfall pada program simpan pinjam koperasi subur Jaya Mandiri subang," *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, vol. 14, no. 1, pp. 44–53, 2019. doi: 10.35969/interkom.v14i1.46.
- [7] A. Aris, R. Anggara, and Z. A. Zamzami, "Perancangan sistem informasi penerimaan siswa baru berbasis web pada PKBM bhakti sejahtera," *CICES*, vol. 2, no. 1, pp. 87–98, 2016. doi: 10.33050/cices.v2i1.215.
- [8] Y. Heriyanto, "Perancangan sistem informasi rental mobil berbasis web pada PT.APM Rent Car," *Jurnal Intra Tech*, vol. 2, no. 2, pp. 64–77, 2018. doi: 10.37030/jit.v2i2.35.
- [9] R. A. Sukamto and M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek (Edisi Revisi)*. Bandung: Informatika, 2018.
- [10] M. Rahmayu, "Rancang Bangun Sistem Informasi Pada Rumah Sakit Dengan Layanan Intranet Menggunakan Metode Waterfall," *Jurnal Evolusi*, vol. 4, no. 2, pp. 33–40, 2016. doi: 10.31294/evolusi.v4i2.699.