

Sistem Pendukung Keputusan Penentuan Bonus Karyawan Berdasarkan Penilaian Kerja Dengan Metode AHP (Analytical Hierarchy Process)
Decision Support System For Determining Employee Bonus Based On Work Assessments With AHP (Analytical Hierarchy Process)

Muhammad Zidane Syach Rizqullah¹⁾, Henny Wahyu Sulistyo²⁾, Nur Qodariyah Fitriyah³⁾

¹Mahasiswa Fakultas Teknik, Universitas Muhammadiyah Jember
email: zidansyach55@gmail.com

²Dosen Fakultas Teknik, Universitas Muhammadiyah Jember
email: heny.sulistyo@unmuhjember.ac.id

³Dosen Fakultas Teknik Universitas Muhammadiyah Jember
email: nurfitriyah@unmuhjember.ac.id

Abstrak

Persaingan dalam pemanfaatan teknologi dalam suatu perusahaan sebenarnya sangat dipengaruhi oleh peran karyawan. Dalam konteks globalisasi dan perkembangan teknologi yang pesat, sebuah perusahaan dituntut untuk terus berinovasi dan memanfaatkan teknologi secara efektif guna tetap bersaing di pasar yang semakin ketat. Oleh karena itu, penting bagi perusahaan untuk memperhatikan pengembangan karyawan dalam hal pengetahuan dan keterampilan teknologi agar dapat memaksimalkan potensi teknologi sebagai alat untuk meningkatkan daya saing dan mencapai keberhasilan bisnis. Perusahaan merupakan salah satu perusahaan yang harus bersaing unggul dalam kualitas karyawan dan teknologi informasi. Namun, Perusahaan masih melakukan penilaian kinerja karyawannya secara manual sehingga proses penilaian akan memakan waktu yang lama dan rentan terjadi kesalahan. Penelitian ini bertujuan untuk meningkatkan pemahaman bonus karyawan di Perusahaan adalah dengan menggunakan metode AHP karena dapat menyediakan kerangka kerja yang terstruktur dan komprehensif untuk mengevaluasi berbagai faktor yang memengaruhi kinerja karyawan secara sistematis. Hasil dari penelitian tersebut nilai dari program Python cenderung lebih tinggi daripada perhitungan manual yang menunjukkan bahwa, metode AHP dalam program Python bisa lebih sensitif terhadap bobot dan kriteria, memberikan hasil yang lebih terstruktur dan berbasis data.

Kata Kunci: Sistem Pendukung Keputusan, AHP dan Penilaian Kerja.

Abstract

Competition in the use of technology in a company is actually greatly influenced by the role of employees. In the context of globalization and rapid technological development, a company is required to continue to innovate and utilize technology effectively to remain competitive in an increasingly tight market. Therefore, it is important for companies to pay attention to employee development in terms of technological knowledge and skills in order to maximize the potential of technology as a tool to increase competitiveness and achieve business success. The company is one of the companies that must compete to excel in employee quality and information technology. However, the Company is still assessing the performance of its employees manually so that the assessment process will take a long time and is prone to errors. This research aims to improve the understanding of employee bonuses in the Company by using the AHP method because it can provide a structured and comprehensive framework for evaluating various factors that affect employee performance systematically. The results of the study the value of the Python program tends to be higher than the manual calculation which shows that, the AHP method in the Python program can be more sensitive to weights and criteria, providing more structured and data-based results.

Keywords: Decision Support System, AHP and Job Appraisal.

1. PENDAHULUAN

Persaingan dalam pemanfaatan teknologi di dalam suatu Perusahaan sebenarnya sangat dipengaruhi oleh peran karyawan. Dalam konteks globalisasi dan perkembangan teknologi yang pesat, perusahaan-perusahaan dituntut untuk terus berinovasi dan memanfaatkan teknologi secara efektif guna tetap bersaing di pasar yang semakin ketat. Namun, efektivitas penggunaan teknologi tidak semata-mata ditentukan oleh tingkat kecanggihan perangkat atau sistem yang dipakai, melainkan juga oleh kemampuan karyawan dalam menerapkan teknologi tersebut. Karyawan tidak hanya sebagai pengguna teknologi, tetapi juga sebagai penggerak utama dalam proses implementasi dan optimalisasi teknologi sesuai dengan kebutuhan dan tujuan perusahaan (Parhusip, 2019). Oleh karena itu, penting bagi perusahaan untuk memperhatikan pengembangan karyawan dalam hal pengetahuan dan keterampilan teknologi agar dapat memaksimalkan potensi teknologi sebagai alat untuk meningkatkan daya saing dan mencapai keberhasilan bisnis.

Perusahaan adalah entitas yang menjadi pusat aktivitas produksi barang atau jasa. Dalam perusahaan, berbagai faktor produksi seperti tenaga kerja, modal, sumber daya alam, dan kewirausahaan berkumpul dan bekerja sama untuk mencapai tujuan bersama. Perusahaan berperan sebagai mediator yang menghubungkan berbagai sumber daya dengan pasar, memastikan bahwa proses produksi berjalan efisien dan efektif untuk mencapai hasil yang optimal (Badu dkk., 2021).

Di beberapa perusahaan, penentuan bonus karyawan masih dilakukan secara manual menggunakan alat seperti Excel. Metode ini memiliki beberapa kekurangan yang signifikan, terutama terkait efisiensi dan akurasi. Penggunaan Excel untuk menentukan bonus sering kali memakan waktu yang lama dan rentan terhadap kesalahan manusia, karena data harus dimasukkan dan dihitung secara manual. Selain itu, metode ini juga mengandung unsur subjektivitas yang tinggi..

Dengan demikian, untuk meningkatkan efisiensi dan akurasi dalam penentuan bonus, perusahaan perlu mempertimbangkan

penggunaan sistem yang lebih canggih dan terotomatisasi. Sistem seperti *Analytical Hierarchy Process* (AHP) dapat membantu mengurangi subjektivitas dan memastikan bahwa penilaian kinerja dan penentuan bonus dilakukan secara lebih obyektif dan konsisten.

Metode *Analytical Hierarchy Process* (AHP) dinilai sebagai pendekatan utama dalam pengembangan data mining dengan pertimbangan keunggulan yang dimilikinya. Salah satu keunggulan utama AHP adalah kemampuannya untuk memungkinkan evaluasi yang sistematis terhadap berbagai alternatif dengan memperhitungkan berbagai faktor yang relevan. Menurut (Firdaus dkk., 2020) AHP adalah model pendukung keputusan yang memecah masalah yang kompleks dan multi kriteria menjadi struktur hirarki.

2. TINJAUAN PUSTAKA

A. Metode AHP dalam Pengambilan Keputusan

Analytical Hierarchy Process (AHP) merupakan salah satu metode pengambilan keputusan multi-kriteria yang dikembangkan oleh Thomas L. Saaty. AHP digunakan untuk mengatasi kompleksitas dalam pengambilan keputusan dengan memecah masalah ke dalam hierarki kriteria dan subkriteria, lalu melakukan perbandingan dua per dua terhadap elemen-elemen tersebut. AHP memiliki beberapa langkah dasar, yaitu:

1. Pembentukan Hierarki: Identifikasi kriteria utama dan subkriteria yang relevan dalam konteks pengambilan keputusan bonus karyawan di perusahaan.
2. Penilaian Berpasangan: Memberikan nilai relatif antar kriteria dan subkriteria berdasarkan tingkat kepentingannya dalam pengambilan keputusan.
3. Konsistensi: Memastikan konsistensi dalam perbandingan berpasangan untuk menghindari bias dan kekeliruan.
4. Perhitungan Bobot: Menghitung bobot relatif untuk setiap kriteria dan subkriteria, mencerminkan tingkat kepentingan relatif.
5. Agregasi dan Pengambilan Keputusan: Menggabungkan bobot untuk mendapatkan nilai total dan mendukung proses pengambilan keputusan.

B. Konsep Perhitungan AHP

Metode Analytic Hierarchy Process (AHP) yang dikembangkan oleh Thomas L. Saaty pada tahun 1970-an adalah teknik yang kompleks dan sistematis untuk mengambil keputusan, terutama ketika keputusan tersebut melibatkan banyak faktor dan kompleksitas. Berikut ini adalah gambaran umum tentang konsep perhitungan dalam AHP (Mustika dkk., 2022).

1. Membuat matriks perbandingan berpasangan: Tentukan kriteria utama (C) dan subkriteria (S) yang relevan. Bentuk matriks perbandingan berpasangan antar elemen dalam hierarki.
2. Berikan nilai perbandingan relatif antar elemen dengan menggunakan skala Saaty, misalnya:
 - a. 1 untuk kesetaraan.
 - b. 3 untuk sedikit lebih penting.
 - c. 5 untuk lebih penting.
 - d. 7 untuk jauh lebih penting.
 - e. 9 untuk sangat lebih penting.
 - f. 2, 4, 6, 8 untuk nilai perbandingan yang berada di antara dua nilai sebelumnya.
3. Konsistensi Hitung nilai eigen (λ) dan vector eigen (v) untuk setiap pasangan elemen. Hitung indeks konsistensi (CI) dengan rumus.

$$CI = \frac{\lambda_{max} - n}{n - 1} \quad (1)$$

dimana n adalah jumlah kriteria. Nilai CR yang kecil menunjukkan konsistensi yang baik. Lalu hitung rasio konsistensi (CR) yaitu membagi CI dengan indeks random (RI) yang sesuai dengan n .

$$CR = \frac{CI}{RI} \quad (2)$$

4. Normalisasi: Normalisasikan vektor eigen (v) untuk mendapatkan bobot relatif (w) dengan rumus:

$$w_i = \frac{v_i}{\sum_{i=1}^n v_i} \quad (3)$$

di mana w_i adalah bobot relatif dari kriteria atau subkriteria ke- i .

5. Agregasi dan Pengambilan Keputusan: Hitung nilai total (v) untuk setiap alternatif dengan mengalikan bobot relatif dengan

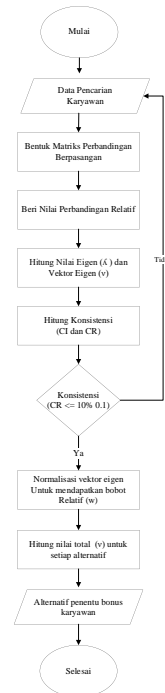
skor atau nilai masing-masing alternatif, dan jumlahkan hasilnya.

$$V = \sum_{i=1}^n (w_i \times \text{nilai alternatif ke-}i) \quad (4)$$

3. METODOLOGI PENELITIAN

A. Perhitungan AHP

Setelah data karyawan dan kriteria penilaiannya didapatkan, langkah selanjutnya adalah proses perhitungan AHP seperti pada flowchart berikut.



Gambar 1. Perhitungan AHP
 Sumber: Penulis 2024

B. Perbandingan Kriteria Berpasangan (Matrices AHP)

Dalam metode AHP, matriks perbandingan berpasangan antar kriteria dan subkriteria perlu dikumpulkan. Ini melibatkan input dari para pengambil keputusan yang memberikan nilai perbandingan relatif antar elemen dalam hierarki. Berikut intensitas kepentingan kriteria berpasangan menurut (Iqbal & Yulianton, 2024).

Tabel 1. Intensitas Kepentingan Matriks Kriteria Berpasangan

Intensitas Kepentingan	Definisi
1	Sama pentingnya dibanding dengan yang lain
3	Sedikit lebih penting dibanding yang lain
5	Cukup penting dibanding yang lain
7	Sangat penting dibanding yang lain
9	Ekstrim pentingnya dibanding yang lain
2, 4, 6, 8	Nilai diantara dua penilaian yang berdekatan
Resiprokal	Jika elemen I memiliki salah satu angka di atas dibandingkan elemen J, maka j memiliki nilai kebalikannya ketika dibanding dengan I.

Sumber: Penulis 2024

Adapun standar nilai konsistensi *random* untuk membobot jumlah kriteria yang digunakan (*Random consistency index*) seperti tabel berikut menurut Dr. Thomas L. Saaty.

Tabel 2. *Random consistency index*

Jumlah Kriteria	<i>Random consistency index (RI)</i>
1	0,00
2	0,00
3	0,58
4	0,90
5	1,12
6	1,24
7	1,32
8	1,41
9	1,45
10	1,49

Sumber: Dr. Thomas L. Saaty

- Sebelum dilakukan perhitungan, perlu ada pembobotan untuk setiap kriteria dan sub-kriteria yang akan digunakan sebagai acuan untuk memilih karyawan yang layak mendapatkan bonus dengan menggunakan kode C1 sampai C5 untuk kriteria dan A1 sampai A4 untuk sub-kriteria.

Tabel 3. Kode Kriteria dan Sub-kriteria

Kode Kriteria	Kriteria	Kode Sub-kriteria	Sub-kriteria
C1	Produktivitas	A1	Pekerjaan selesai > 90%
		A2	Pekerjaan selesai 80 - 90 %
		A3	Pekerjaan selesai 60 - 79%
		A4	Pekerjaan selesai < 60 %
C2	Komunikasi dan Kerjasama	A5	Sangat Baik
		A6	Baik
		A7	Kurang
		A8	Tidak Mampu
C3	Pemahaman SOP/SPK	A9	Sangat Mampu
		A10	Mampu
		A11	Kurang Mampu
		A12	Tidak Mampu
C4	Pemahaman Tools / Software / Alat Bantu Kerja	A13	Pemahaman Baik
		A14	Pemahaman Kurang Baik
		A15	Tidak Memahami
C5	Kedisiplinan	A16	Tidak Ada Pelanggaran
		A17	Sedikit Pelanggaran
		A18	Banyak Pelanggaran

Sumber: Penulis 2024

- Langkah kedua yaitu menghitung matriks perbandingan kriteria atau matriks AHP dari kriteria untuk menemukan keterkaitan antar kriteria yang digunakan.

Tabel 4. Perbandingan Kriteria Berpasangan

Kriteria	C1	C2	C3	C4	C5
C1	1	5	4	7	5
C2	1/5	1	3	4	2
C3	1/4	1/3	1	3	2
C4	1/7	1/4	1/3	1	3
C5	1/5	1/2	1/2	1/3	1

Sumber: Penulis 2024

Lalu hitung total kolom untuk memudahkan normalisasi

Tabel 5. Hasil Matriks Perbandingan Kriteria Berpasangan

Kriteria	C1	C2	C3	C4	C5	Total
C1	1	5	4	7	5	22
C2	0.2	1	3	4	2	10.2
C3	0.25	0.33	1	3	2	6.583
C4	0.14	0.25	0.33	1	3	4.720
C5	0.2	0.5	0.5	0.33	1	2.533
Total	1.79	7.08	8.83	15.33	13.00	46.033

Sumber: Penulis 2024

- Selanjutnya mencari prioritas dan *eigen value* untuk menentukan *Consistency Rate*. Sebelum itu nilai harus dinormalisasikan terlebih dahulu dengan cara membagi nilai di setiap kriteria dengan hasil jumlah pada tiap kolom.

Tabel 6. Normalisasi

Kriteria	C1	C2	C3	C4	C5	Total
C1	1/1.79	5/7.08	4/8.83	7/15.33	5/13.00	0.559 + 0.706 + 0.453 + 0.456 + 0.385 = 2.559
C2	0.2/1.79	1/7.08	3/8.83	4/15.33	2/13.00	0.112 + 0.141 + 0.340 + 0.261 + 0.154 = 1.008
C3	0.25/1.79	0.33/7.08	1/8.83	3/15.33	2/13.00	0.140 + 0.047 + 0.113 + 0.196 + 0.154 = 0.650
C4	0.14/1.79	0.25/7.08	0.33/8.83	1/15.33	3/13.00	0.078 + 0.035 + 0.037 + 0.065 + 0.231 = 0.446
C5	0.2/1.79	0.5/7.08	0.5/8.83	0.33/15.33	1/13.00	0.112 + 0.070 + 0.057 + 0.022 + 0.077 = 0.338

Sumber: Penulis 2024

Jumlahkan elemen-elemen pada setiap baris, kemudian bagi dengan jumlah kriteria (n=5) untuk mendapatkan bobot relatif atau prioritas.

Tabel 7. Bobot Relatif

Kriteria	Total	Prioritas (Bobot Relatif)
C1	2.559	2.559 / 5 = 0.512
C2	1.008	1.008 / 5 = 0.202
C3	0.650	0.650 / 5 = 0.130
C4	0.446	0.446 / 5 = 0.089
C5	0.338	0.338 / 5 = 0.067

Sumber: Penulis 2024

- Selanjutnya adalah menghitung *eigen value* dengan mengalikan jumlah elemen normalisasi setiap baris dengan bobot relatifnya, lalu menjumlahkan hasilnya.

Tabel 8. Perhitungan *Eigen value*

Kriteria	Total	Prioritas	Total * Prioritas
C1	2.559	0.512	2.559* 0.512= 1.309
C2	1.008	0.202	1.008* 0.202= 0.204
C3	0.650	0.130	0.650* 0.130= 0.085
C4	0.446	0.089	0.446* 0.089= 0.040
C5	0.338	0.067	0.338* 0.067= 0.023

Sumber: Penulis 2024

λ max didapat dari total *eigen value*:

$$\lambda \text{ max} = 1.309 + 0.204 + 0.085 + 0.040 + 0.023$$

$$= (1.661/5 = 0,331)$$

- Langkah selanjutnya adalah menghitung konsistensi dengan rumus yang sudah diuraikan sebelumnya. Sehingga nilai CI adalah sebagai berikut

$$CI = \frac{\lambda_{\text{max}} - n}{n - 1} = \frac{0,331 - 5}{4} = -0.919$$

Lalu, rasio konsistensi didapat dari hasil bagi CI dengan RI untuk 5 kriteria adalah 1.12.

$$CR = \frac{CI}{RI} = \frac{-0.919}{1.12} = -0.82$$

4. HASIL DAN PEMBAHASAN

A. Praproses Data

Salah satu langkah penting dalam praproses data adalah menghilangkan atribut atau fitur yang tidak relevan atau tidak diperlukan untuk analisis atau pemodelan yang akan dilakukan. Dalam hal ini, dilakukan penghapusan atribut divisi sehingga dataset akan berbentuk seperti pada tabel di bawah.

Tabel 9. Hasil Praproses Data

Nama	Productivity	Komunikasi	Sop	Tools	Kedisiplinan
Karyawan 1	45	30	15	15	30
Karyawan 2	30	20	15	15	30
Karyawan 3	45	30	15	15	30
Karyawan 4	45	30	15	15	30
Karyawan 5	45	30	15	15	30

Sumber: Penulis 2024

B. Implementasi Program.

1. Import Dataset.

Langkah pertama dalam proses pengujian adalah melakukan upload atau impor dataset menggunakan library pandas. Gambar di bawah merupakan tampilan dataset sebelum melewati proses perhitungan.

Tabel 10. Implementasi *Import Dataset*

Membaca dataset dari file CSV lalu menampilkan Sebagai datanya
1. <code>df= pd.read_excel('AHP5C.xlsx')</code>
2. <code>df.head()</code>

Sumber: Penulis 2024

2. Perbandingan Kriteria Berpasangan.

Langkah selanjutnya adalah mengimplementasikan perbandingan kriteria berpasangan ke dalam program dengan cara membuat sebuah matriks. Setiap matriks direpresentasikan dengan menggunakan angka-angka perbandingan seperti 1/3, 1/4, 1/5, dan sebagainya, Pembuatan matriks ini dilakukan terhadap kriteria dan seluruh sub-kriteria.

Tabel 11. Implementasi Matriks Kriteria Berpasangan

Membuat matriks kriteria berpasangan
1. <code>criteria_comparison = np.array([</code>
2. <code>[1, 5, 4, 7, 5],</code>
3. <code>[1/5, 1, 3, 4, 2],</code>
4. <code>[1/4, 1/3, 1, 3, 2],</code>
5. <code>[1/7, 1/4, 1/3, 1, 3],</code>

Membuat matriks kriteria berpasangan
6. <code>[1/5, 1/2, 1/2, 1/3, 1]</code>
7. <code>])</code>
8.
9. <code>productivity_comparison =</code>
<code>np.array([</code>
10. <code>[1, 3, 5, 9],</code>
11. <code>[1/3, 1, 4, 6],</code>
12. <code>[1/5, 1/4, 1, 3],</code>
13. <code>[1/9, 1/6, 1/3, 1]</code>
14. <code>])</code>
15.
16. <code>communication_comparison =</code>
<code>np.array([</code>
17. <code>[1, 3, 5, 7],</code>
18. <code>[1/3, 1, 3, 5],</code>
19. <code>[1/5, 1/3, 1, 3],</code>
20. <code>[1/7, 1/5, 1/3, 1]</code>
21. <code>])</code>
22.
23. <code>sop_comparison = np.array([</code>
24. <code>[1, 4, 6, 8],</code>
25. <code>[1/4, 1, 3, 5],</code>
26. <code>[1/6, 1/3, 1, 3],</code>
27. <code>[1/8, 1/5, 1/3, 1]</code>
28. <code>])</code>
29.
30. <code>tools_comparison = np.array([</code>
31. <code>[1, 5, 9],</code>
32. <code>[1/5, 1, 3],</code>
33. <code>[1/9, 1/3, 1]</code>
34. <code>])</code>
35.
36. <code>discipline_comparison =</code>
<code>np.array([</code>
37. <code>[1, 5, 8],</code>
38. <code>[1/5, 1, 3],</code>
39. <code>[1/8, 1/3, 1]</code>
40. <code>])</code>

Sumber: Penulis 2024

3. Perhitungan Bobot Kriteria dan Sub-Kriteria.

Selanjutnya dilakukan perhitungan bobot untuk kriteria dan subkriteria menggunakan metode Matriks Perbandingan Berpasangan dengan bantuan library NumPy. Proses ini dimulai dengan menghitung total dari tiap kolom dalam matriks perbandingan untuk mendapatkan nilai relatif antar elemen.

Tabel 12. Implementasi Perhitungan Bobot Kriteria dan Sub-Kriteria

Fungsi Perhitungan Bobot Kriteria dan Sub-Kriteria	
1.	<code>def calculate_weights(matrix):</code>
2.	<code> column_sums = np.sum(matrix,</code> <code> axis=0)</code>
3.	<code> normalized_matrix = matrix /</code> <code> column_sums</code>
4.	<code> row_totals =</code> <code> np.sum(normalized_matrix,</code> <code> axis=1)</code>
5.	<code> priorities = row_totals /</code> <code> len(matrix)</code>
6.	<code> return priorities</code>
7.	<code>criteria_weights =</code> <code> calculate_weights(criteria_comp</code> <code> arison)</code>
8.	<code>productivity_weights =</code> <code> calculate_weights(productivity_</code> <code> comparison)</code>
9.	<code>communication_weights =</code> <code> calculate_weights(communication</code> <code> _comparison)</code>
10.	<code>sop_weights =</code> <code> calculate_weights(sop_comparison</code> <code>)</code>
11.	<code>tools_weights =</code> <code> calculate_weights(tools_comparis</code> <code> on)</code>
12.	<code>discipline_weights =</code> <code> calculate_weights(discipline_co</code> <code> mparison)</code>

Sumber: Penulis 2024

Tabel berikut merupakan hasil dari fungsi perhitungan yang akan digunakan sebagai bobot alternatif untuk penentuan peringkat. Dimana C

dan A merupakan kode dari kriteria dan sub-kriteria.

Tabel 13. Hasil Implementasi Pehitungan Bobot Kriteria dan Sub-Kriteria

Kode Kriteria	Kriteria	Kode Sub-kriteria	Sub-kriteria
C1	0,511	A1	0,561
		A2	0,283
		A3	0,108
		A4	0,048
C2	0,20	A5	0,558
		A6	0,263
		A7	0,122
		A8	0,057
C3	0,13	A9	0,606
		A10	0,232
		A11	0,110
C4	0,09	A12	0,052
		A13	0,748
C5	0,07	A14	0,180
		A15	0,071
		A16	0,737
		A17	0,186
		A18	0,077

Sumber: Penulis 2024

4. Penentuan Bobot Relatif Terhadap Nilai dalam Dataset

Pada tahap ini, dilakukan penentuan bobot relatif terhadap nilai dalam dataset berdasarkan kriteria yang telah ditetapkan. Melalui kode yang disediakan, setiap nilai subkriteria seperti produktivitas, komunikasi, SOP, tools, dan kedisiplinan dievaluasi untuk menentukan bobotnya.

Tabel 14. Implementasi Penentuan Bobot Kriteria dan Sub-Kriteria

Fungsi Perhitungan Bobot Kriteria dan Sub-Kriteria	
1.	<code>def</code> <code> get_subcriteria_weight(criteria</code> <code> , value):</code>
2.	<code> if criteria ==</code> <code> 'productivity':</code>
3.	<code> if value == 60:</code>
4.	<code> return</code> <code> productivity_weights[0]</code>
5.	<code> elif value == 45:</code>
6.	<code> return</code> <code> productivity_weights[1]</code>

```

Fungsi Perhitungan Bobot Kriteria dan Sub-Kriteria
7.         elif value == 30:
8.             return
           productivity_weights[2]
9.         else:
10.            return
            productivity_weights[3]
11.        elif criteria ==
            'komunikasi':
12.            if value == 40:
13.                return
                communication_weights[0]
14.            elif value == 30:
15.                return
                communication_weights[1]
16.            elif value == 20:
17.                return
                communication_weights[2]
18.        else:
19.            return
            communication_weights[3]
20.        elif criteria == 'SOP':
21.            if value == 20:
22.                return sop_weights[0]
23.            elif value == 15:
24.                return sop_weights[1]
25.            elif value == 10:
26.                return sop_weights[2]
27.        else:
28.            return sop_weights[3]
29.        elif criteria == 'tools':
30.            if value == 20:
31.                return sop_weights[0]
32.            elif value == 15:
33.                return sop_weights[1]
34.            elif value == 10:
35.                return sop_weights[2]
36.        else:
37.            return sop_weights[3]
38.        elif criteria ==
            'kedisiplinan':
39.            if value == 40:
    
```

```

Fungsi Perhitungan Bobot Kriteria dan Sub-Kriteria
40.        return sop_weights[0]
41.        elif value == 30:
42.            return sop_weights[1]
43.        elif value == 20:
44.            return sop_weights[2]
45.        else:
46.            return sop_weights[3]
    
```

Sumber: Penulis 2024

Proses ini memastikan bahwa setiap elemen dalam dataset memiliki kontribusi yang sesuai dalam penilaian akhir, yang kemudian digunakan untuk analisis lebih lanjut atau pengambilan keputusan berdasarkan prioritas kriteria yang telah diatur sebelumnya. Langkah selanjutnya adalah menerapkan fungsi di atas ke dataset seperti tabel di bawah ini.

Tabel 15. Implementasi Fungsi kedalam Dataset

```

Menerapkan fungsi kedalam dataset
1. for criteria in ['productivity',
    'komunikasi', 'SOP', 'tools',
    'kedisiplinan']:
2. df[f'{criteria}_weight'] =
    df[criteria.apply(lambda x:
        get_subcriteria_weight(criteria
        , x))
    
```

Sumber: Penulis 2024

Gambar di bawah ini merupakan bentuk dataset setelah fungsi di atas diterapkan. Nilai dataset kini berubah menjadi bobot prioritas sub-kriteria yang siap diperingkatkan.

	nama	productivity_weight	komunikasi_weight	SOP_weight	tools_weight	kedisiplinan_weight
0	Karyawan 1	0.283001	0.263345	0.231831	0.231831	0.231831
1	Karyawan 2	0.108224	0.121873	0.231831	0.231831	0.231831
2	Karyawan 3	0.283001	0.263345	0.231831	0.231831	0.231831
3	Karyawan 4	0.283001	0.263345	0.231831	0.231831	0.231831
4	Karyawan 5	0.283001	0.263345	0.231831	0.231831	0.231831
5	Karyawan 6	0.283001	0.263345	0.231831	0.231831	0.231831
6	Karyawan 7	0.108224	0.263345	0.231831	0.231831	0.231831
7	Karyawan 8	0.108224	0.056890	0.052077	0.231831	0.231831
8	Karyawan 9	0.108224	0.056890	0.231831	0.231831	0.231831
9	Karyawan 10	0.108224	0.121873	0.231831	0.231831	0.052077
10	Karyawan 11	0.047548	0.056890	0.231831	0.231831	0.231831
11	Karyawan 12	0.108224	0.056890	0.231831	0.231831	0.110398
12	Karyawan 13	0.283001	0.263345	0.231831	0.231831	0.231831
13	Karyawan 14	0.283001	0.121873	0.052077	0.231831	0.110398
14	Karyawan 15	0.047548	0.056890	0.110398	0.056890	0.056890
15	Karyawan 16	0.047548	0.263345	0.056890	0.231831	0.110398
16	Karyawan 17	0.047548	0.557892	0.056890	0.110398	0.110398
17	Karyawan 18	0.108224	0.056890	0.052077	0.056890	0.110398
18	Karyawan 19	0.047548	0.557892	0.052077	0.231831	0.110398
19	Karyawan 20	0.581227	0.121873	0.052077	0.110398	0.052077

Gambar 2. Penentuan Bobot Sub-Kriteria Kedisiplinan

Sumber: Penulis 2024

5. **Pemeringkatan Berdasarkan Skor Akhir**
 Selanjutnya, hasil skor tersebut digunakan untuk menentukan peringkat karyawan serta klasifikasi *grade* dan besaran bonus yang diberikan, yang mencerminkan tingkat pencapaian mereka dalam konteks kriteria yang telah diatur sebelumnya.

Tabel 16. Pemeringkatan Skor

```

Fungsi menghitung skor akhir dan pemeringkatan
1. def calculate_total_score(row):
2.     subcriteria_weights = [
3.
4.         row['productivity_weight'],
5.         row['komunikasi_weight'],
6.         row['SOP_weight'],
7.         row['tools_weight'],
8.         row['kedisiplinan_weight']
9.     ]
10.    total_score = sum([cw * sw
11.                       for cw, sw in
12.                           zip(criteria_weights,
13.                               subcriteria_weights)])
14.    return total_score
15.
16. df['total_score'] =
17.     df.apply(calculate_total_score,
18.              axis=1)
19.
20. def
21.     determine_grade_and_bonus(total
22.         _score):
23.     if total_score <= 0.105:
24.         return 'D', '25% dari
25.             gaji pokok/bulan'
26.     elif 0.105 < total_score <=
27.         0.140:
28.         return 'C', '50% dari
29.             gaji pokok/bulan'
30.     elif 0.140 < total_score <=
31.         0.170:
    
```

```

20.         return 'B', '75% dari
21.             gaji pokok/bulan'
22.     elif total_score > 0.170:
23.         return 'A', '100% dari
24.             gaji pokok/bulan'
25.     else:
26.         return 'Unknown',
27.             'Unknown'
    
```

Sumber: Penulis 2024

C. Hasil Implementasi.

Hasil implementasi program ini terwujud dalam sebuah tabel yang memuat nama karyawan, total skor yang dihitung berdasarkan bobot subkriteria dan kriteria, serta *grade* dan besaran bonus yang sesuai dengan pencapaian masing-masing karyawan. Tabel ini memberikan gambaran yang jelas tentang evaluasi kinerja karyawan berdasarkan dataset yang telah diproses dan dapat memberikan penilaian yang lebih efisien dan objektif dibanding penilaian manual. Efisiensi penilaian ini berdasarkan waktu yang diperlukan untuk melakukan perhitungan penentuan *grade* dan bonus yang didapatkan.

Tabel 17. Hasil Implementasi Program

Nama	Skor_Akhir	Grade	Bonus
Karyawan 1	0,264353	A	100% dari gaji pokok/bulan
Karyawan 2	0,146456	B	75% dari gaji pokok/bulan
Karyawan 3	0,264353	A	100% dari gaji pokok/bulan
Karyawan 4	0,264353	A	100% dari gaji pokok/bulan
Karyawan 5	0,264353	A	100% dari gaji pokok/bulan
Karyawan 6	0,264353	A	100% dari gaji pokok/bulan
Karyawan 7	0,174950	A	100% dari gaji pokok/bulan
Karyawan 8	0,110028	C	50% dari gaji pokok/bulan
Karyawan 9	0,133367	C	50% dari gaji pokok/bulan
Karyawan 10	0,134326	C	50% dari gaji pokok/bulan
Karyawan 11	0,102331	D	25% dari gaji pokok/bulan
Karyawan 12	0,125173	C	50% dari gaji pokok/bulan
Karyawan 13	0,264353	A	100% dari gaji pokok/bulan
Karyawan 14	0,204325	A	100% dari gaji pokok/bulan
Karyawan 15	0,145343	B	75% dari gaji pokok/bulan
Karyawan 16	0,184262	A	100% dari gaji pokok/bulan
Karyawan 17	0,232690	A	100% dari gaji pokok/bulan

Nama	Skor_Akhir	Grade	Bonus
Karyawan 18	0,135384	C	50% dari gaji pokok/bulan
Karyawan 19	0,171705	A	100% dari gaji pokok/bulan
Karyawan 20	0,331811	A	100% dari gaji pokok/bulan

Sumber: Penulis 2024

Tabel 18. Daftar Perbedaan Nilai

Nama	Skor Akhir Program	Grade program	Skor Akhir Manual	Grade Manual
Karyawan 2	0.146456	B	0.139	C
Karyawan 7	0.174950	A	0.168	B
Karyawan 8	0.110028	C	0.103	D
Karyawan 11	0.102331	D	0.095	D
Karyawan 14	0.204325	A	0.256	A
Karyawan 15	0.145343	B	0.169	B
Karyawan 16	0.184262	A	0.235	A
Karyawan 17	0.232690	A	0.244	A
Karyawan 18	0.135384	C	0.154	B
Karyawan 19	0.171705	A	0.224	A
Karyawan 20	0.331811	A	0.340	A

Berdasarkan hasil 11 data yang menunjukkan perbedaan antara penghitungan menggunakan metode AHP dari program Python pada Tabel 4.8 dengan penghitungan manual pada Tabel 4.9. Dari perbedaan tersebut, beberapa nilai yang dihasilkan dari program Python lebih tinggi dan beberapa lebih rendah jika dibandingkan dengan perhitungan manual. Contohnya, karyawan 2 memiliki nilai lebih tinggi pada perhitungan Python (0.146456) dibandingkan dengan perhitungan manual (0.139), tetapi memiliki grade yang lebih rendah. Sedangkan karyawan 7 memiliki grade lebih tinggi pada perhitungan Python dibandingkan perhitungan manual, meskipun nilai totalnya lebih rendah.

5. KESIMPULAN DAN SARAN

A. Kesimpulan

1. Sistem pendukung keputusan menggunakan metode AHP untuk penilaian kinerja karyawan di Perusahaan telah berhasil diimplementasikan ke dalam program *Python*.
2. Dari total 20 data, terdapat 11 data yang menunjukkan perbedaan antara hasil perhitungan metode AHP dari program Python dan perhitungan manual yang dilakukan oleh perusahaan. Nilai dari program *Python* cenderung lebih tinggi

daripada perhitungan manual yang menunjukkan bahwa metode AHP dalam program *Python* terbukti lebih efisien dan objektif dalam menentukan bonus karyawan.

B. Saran

1. Lakukan uji coba lebih lanjut terhadap model yang dikembangkan untuk memastikan validitasnya dengan cara mengumpulkan lebih banyak data dari berbagai sumber.
2. Selain penghitungan skor dan pemberian grade, penelitian selanjutnya dapat mengeksplorasi lebih dalam dampak dari penerapan sistem ini terhadap produktivitas dan motivasi karyawan, serta dampaknya terhadap kinerja keseluruhan perusahaan.

6. REFERENSI

A. Artikel

- Badu, P., Djou, S. R., & Rahman, R. (2021). Analisis Pengelolaan Eksposur Translasi Pada Perusahaan Multinasional (Yamaha Motor Company). *Valuasi (Jurnal Ilmiah Ilmu Manajemen dan Kewirausahaan)*, 1(2), 479–493. <https://doi.org/10.46306/vls.v1i2>.
- Firdaus, A., Rahmat, M., & Wibowo, A. (2020). Sistem Pendukung Keputusan Penilaian Kinerja Pegawai SMA Menggunakan Metode AHP. *JIKA (Jurnal Informatika)*, 4(3), 73–77.
- Fitriyah, N. Q., Oktavianto, H., & Hasbullah, H. (2020). Deteksi Spam Pada Email Berbasis Fitur Konten Menggunakan Naïve Bayes. *JUSTINDO (Jurnal Sistem dan Teknologi Informasi Indonesia)*, 5(1), 1-7.
- Fitriyah, N. Q., Sulisty, H. W., & Arifin, Z. (2023). Peningkatan Kinerja Staf Desa Sumbersalak Melalui Pelatihan Dasar Microsoft Word. *Journal Of Humanities Community Empowerment*, 1(1), 19-24.
- Fitriyah, N., Warsito, B., & Di Asih, IM (2020). Analisis Sentimen Gojek Pada Media Sosial Twitter Dengan Klasifikasi Support Vector Machine (SVM). *Jurnal Gaussian*,

9 (3), 376-390.

- Iqbal, M., & Yulianton, H. (2024). Penerapan Metode AHP Dalam Sistem Pendukung Keputusan Pemilihan Pupuk Efisien Untuk Budidaya Bawang Merah (Studi Kasus Kabupaten Demak). *JOUTICA (Journal of Informatic)*
- Mustika, Lisawita, Yunefri, Y., & Costaner, L. (2022). Sistem Pendukung Keputusan Penilaian Kinerja Pegawai Menggunakan Metode Ahp (Studi Kasus Puskesmas Sapta Taruna kecamatan bukit raya pekanbaru). *JURKIM (Jurnal Karya Ilmiah Multidisiplin)*, 2(1), 1–10.
- Parhusip, J. (2019). Penerapan Metode Analytical Hierarchy Process (AHP) Pada Desain Sistem Pendukung Keputusan Pemilihan Calon Penerima Bantuan Pangan Non Tunai (BPNT) Di Kota Palangka Raya. *Jurnal Teknik Informatika*, 13(2), 18–29.
- Sulistyo, H. W. (2017). Evaluasi Penerimaan Mahasiswa Terhadap Sistem Informasi Akademik (SIA) Menggunakan TAM 3 Di Universitas Muhammadiyah Jember (Doctoral dissertation, Universitas Islam Indonesia).
- Sulistyo, H. W., & Oktavianto, H. (2020). Perancangan Dan Implementasi File Sharing Menggunakan Samba Server. *Jurnal Aplikasi Sistem Informasi Dan Elektronika*, 2(1), 24-30.
- Wibisono, W., & Baskoro, F. (2002). Pengujian Perangkat Lunak Dengan Menggunakan Model Behaviour UML. *JUTI (Jurnal Ilmiah Teknologi Informasi)*, 1(1), 43–50.