



IJIS - Indonesian Journal on Information System

eISSN : 2548-6438

pISSN : 2614-7173



Di terbitkan oleh Lembaga Penelitian dan Pengabdian Masyarakat
Politeknik Sains dan Teknologi Widyadarmas
Maluku Utara 2016



Daftar Isi

Artikel

SISTEM INFORMASI MONITORING AKADEMIK SISWA PADA SMAN 9 BALIKPAPAN DENGAN EXTREME PROGRAMMING

PDF / (INDONESIA)
1-13

 Abstract views : 61 times

 Yuyun Tri Wiranti, Sri Rahayu Natasia, Vika Fitratunnany Insanittaqwa, Rendy Setyawan

 <https://doi.org/10.36549/ijjs.v7i1.195>

IMPLEMENTASI METODE CERTAINTY FACTOR PADA SISTEM PAKAR DIAGNOSIS GANGGUAN GIZI ANAK BALITA BERBASIS MOBILE

PDF / (INDONESIA)
14-24

 Abstract views : 71 times

 Salman Pigi, Yogo Dwi Prasetyo, Amalia Beladinna Arifa

 <https://doi.org/10.36549/ijjs.v7i1.203>

SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB PADA SDN CAWANG 12 PAGI

PDF / (INDONESIA)
25-34

 Abstract views : 82 times

 Aditiya Dwi Pangestu, Lilyani Asri Utami

 <https://doi.org/10.36549/ijjs.v7i1.196>

SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN BAHASA PEMROGRAMAN DENGAN METODE ANALYTICAL HIERARCHY PROCESS

PDF / (INDONESIA)
35-48

 Abstract views : 89 times

 Endang Pujiastuti, Lia Mazia, Annissa Mareti, Ade Apriliana, Anggita Ayu Nandasari

 <https://doi.org/10.36549/ijjs.v7i1.205>

MEMBANGUN E-GOVERMENT SISTEM INFORMASI PEMERINTAH DESA BERBASIS WEB DI PULAU MAITARA

PDF / (INDONESIA)
49-58

 Abstract views : 69 times

 Yuslinah Nurdiani, Adelina Ibrahim, Arisandy Ambarita, Saiful Do Abdullah

 <https://doi.org/10.36549/ijjs.v7i1.191>

PERANCANGAN SISTEM INFORMASI BANK SAMPAH BERBASIS WEBSITE

PDF / (INDONESIA)
59-70

 Abstract views : 115 times

 S Atin, S Mutia, A Widayanti, H S Yatawa, A A Rafdhi, I Afrianto

 <https://doi.org/10.36549/ijjs.v7i1.194>

IMPLEMENTASI ALGORITMA FP-GROWTH UNTUK MENEMUKAN POLA PEMBELIAN KONSUMEN PADA ANALISIS KERANJANG PASAR

PDF / (INDONESIA)
71-78

 Abstract views : 68 times

 Lusa Indah Prahartiwi

 <https://doi.org/10.36549/ijjs.v7i1.208>

PEMILIHAN SISWA BERPRESTASI DENGAN MENERAPKAN METODE ANALYTICAL HIERARCHY PROCESS UNTUK SISTEM PENDUKUNG KEPUTUSAN (STUDI KASUS : SEKOLAH MENANGAH ATAS YAPERMAS)

PDF / (INDONESIA)
79-89

 Abstract views : 100 times

 Ari Puspita, Muhammad Fahmi, Yuyun Yuningsih, Mohammad Haddiel Fuad

 <https://doi.org/10.36549/ijjs.v7i1.207>

**SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN BAHASA
PEMROGRAMAN DENGAN METODE
ANALYTICAL HIERARCHY PROCESS**

**PROGRAMMING LANGUAGE SELECTION DECISION SUPPORT
SYSTEM WITH ANALYTICAL HIERARCHY PROCESS METHOD**

Endang Pujiastuti¹, Lia Mazia², Annissa Maret³, Ade Apriliana⁴, Anggita Ayu Nandasari⁵
Fakultas Teknologi Informasi, Program Studi Sistem Informasi, Fakultas Ekonomi dan
Bisnis, Program Studi Bisnis Digital
Universitas Nusa Mandiri
Email: endang.epj@nusamandiri.ac.id

Abstrak

Penggunaan web dinilai sangat berguna dan efektif untuk menyebarkan informasi terlihat dari meningkatnya permintaan perusahaan atau institusi akan jasa pembuatan sebuah web yang biasa dikerjakan oleh seorang programmer. Dalam pembuatan web membutuhkan peran bahasa pemrograman yang digunakan oleh programmer. Bahasa pemrograman merupakan bahasa yang terdiri dari sintaks-sintaks yang didokumentasikan agar menjadi sebuah web. Penggunaan bahasa pemrograman tersebut menimbulkan kebutuhan untuk membuat suatu pemilihan keputusan oleh programmer untuk menentukan bahasa pemrograman mana yang tepat untuk digunakan sesuai dengan kebutuhannya. Penelitian ini menggunakan metode *AHP* yang menggunakan alternatif dan kriteria untuk membuat struktur hierarki agar dapat memudahkan proses pengambilan keputusan. Berdasarkan hasil perhitungan dari lima bahasa pemrograman yang dipilih, ada *JavaScript* dengan hasil perhitungan sebesar 32,68% menjadi bahasa pemrograman paling banyak digunakan dan berada di tingkat atas pilihan dari setiap kriteria kemudian bahasa lainnya ada *PHP*, *Python*, *C++* dan *Java*, dengan kriteria bahasa pemrograman sederhana/mudah dipelajari menjadi kriteria paling banyak dipilih dengan persentase tertinggi dari kriteria *powerfull*, populer, dan *portable/cross platform*.

Kata Kunci: Bahasa Pemrograman, *AHP*, Sistem Pendukung Keputusan.

Abstract

The web is considered very useful and practical for disseminating information, as seen from the increasing demand for companies or institutions for web creation services which a programmer usually does. Making the web requires the role of the programming language used by the programmer. Programming language is a

language that consists of syntaxes that are documented to become a web. Using the programming language raises the need to decide by the programmer to determine which programming language is appropriate according to his needs. This study uses the AHP method that uses alternatives and criteria to create a hierarchical structure to facilitate decision-making. Based on the results of calculations from the five selected programming languages, there is JavaScript. The calculation result of 32.68% is the most widely used programming language and is at the top level of choice from each criterion. The other languages are PHP, Python, C++, and Java, with programming language criteria that are simple/easy to learn. The most chosen criteria with the highest percentage are powerful, popular, and portable/cross-platform.

Keywords: Programming Language, AHP, Decision Support System

PENDAHULUAN

Pada era digital sekarang ini, hampir semua orang menghabiskan waktunya untuk mengakses internet, mulai dari kegiatan kecil semacam mencari hiburan, informasi, sampai mengurus pekerjaan. Tersedia banyak media yang bisa menjadi wadah informasi yang biasa diakses oleh masyarakat saat ini, salah satunya melalui sebuah web. Penggunaan web dinilai sangat berguna dan efektif untuk menyebarkan informasi terlihat dari meningkatnya permintaan perusahaan atau institusi akan jasa pembuatan sebuah web yang biasa dikerjakan oleh seorang *programmer* [1].

Dalam pembuatan web membutuhkan peran bahasa pemrograman yang digunakan oleh *programmer*. Bahasa pemrograman merupakan bahasa yang terdiri dari sintaks-sintaks yang didokumentasikan agar menjadi sebuah web. Bahasa pemrograman menjadi sarana untuk

mengimplemantasikan solusi dari permasalahan algoritmik [2]. Menentukan bahasa pemrograman dibutuhkan karena mempengaruhi faktor dalam proses pembuatan web agar tepat dan sesuai dengan kebutuhan. Bahkan penguasaan bahasa pemrograman tertentu menjadi sebuah kualifikasi yang bernilai dalam diri seorang *programmer*. Saat ini banyak bahasa pemrograman yang dapat dipahami dan banyak digunakan oleh para *programmer* profesional maupun yang masih pemula dalam hal ngoding.

Terdapat 10 bahasa pemrograman populer di Indonesia yaitu *Java*, Bahasa *C*, *PHP*, *Visual Basic*, *Phyton*, *C++*, *JavaScript*, *C#*, *Objective-C*, dan *ActionScript*. Walau pun terdapat banyak bahasa pemrograman yang beragam namun masing-masing mempunyai kelebihan dan kekurangannya [3]. Penggunaan bahasa pemrograman

tersebut menimbulkan kebutuhan untuk membuat suatu pemilihan keputusan oleh programmer untuk menentukan bahasa pemrograman mana yang tepat untuk digunakan sesuai dengan kebutuhannya.

Untuk itu penelitian ini membutuhkan metode untuk menunjang dalam membuat keputusan seperti permasalahan pemilihan bahasa pemrograman ini. Metode *Analytical Hierarchy Process* (AHP) merupakan salah satu metode dalam sebuah sistem untuk menunjang pengambilan keputusan dengan menggunakan pertimbangan sebuah perbandingan dan fleksibilitas terhadap permasalahan yang bersifat multikriteria sehingga menghasilkan kriteria dan alternatif. Metode ini membuat suatu permasalahan multikriteria memiliki struktur hierarki. Dengan struktur hierarki, suatu masalah dapat terlihat sistematis dan lebih terstruktur karena masalah yang dihadapi dapat diuraikan menjadi beberapa bagian dan bagian tersebut disusun ke dalam struktur hierarki [4].

LANDASAN TEORI

Thomas L. Saaty merupakan pengembang metode pendukung keputusan *Analytical Hierarchy Process* pada tahun 1970-an. Metode pendukung keputusan ini menggambarkan permasalahan kompleks menjadi beberapa kelompok dalam struktur hierarki sehingga menjadi lebih terstruktur dan

sistematis, dimana tingkatan pertama yaitu tujuan, kemudian tingkatan faktor, kondisi, sub-ketentuan, dan seterusnya hingga pada tingkat pemilihan terakhir [27].

Kelebihan AHP adalah mengambil keputusan dengan membandingkan setiap kriteria yang terkait dengan masalah secara berpasangan, sehingga ada nilai bobot kepentingan dari setiap kriteria yang ada [28]. AHP memberikan kemampuan untuk menyelesaikan masalah dengan berbagai kondisi dan kondisi lainnya dengan membandingkan preferensi setiap elemen dalam hierarki. Oleh karena itu, AHP dapat dikatakan merupakan metode pengambilan keputusan yang kompleks [29].

Terdapat beberapa prinsip yang perlu dipahami untuk menyelesaikan masalah dengan *Analytical Hierarchy Process*, yaitu:

1. *Decomposition* (membuat hierarki)

Membuat bagian secara hierarki dengan membagi sebuah permasalahan kompleks menjadi bagian lebih kecil supaya dapat dimengerti dengan mudah.

2. *Comparative Judgment* (evaluasi kriteria serta alternatif)

Perbandingan berpasangan dilakukan untuk mengevaluasi

kriteria dan alternatif, skala terbaik untuk mengekspresikan pendapat adalah skala 1 sampai 9. Tabel analisis dipakai untuk mengukur bobot serta arti pendapat kuantitatif dari skala perbandingan yang dapat digunakan, antara lain:

Tabel 1. Nilai Perbandingan Berpasangan

Intensitas Kepentingan	Keterangan
1	Kedua elemen sama pentingnya
3	Elemen yang satu sedikit lebih penting daripada elemen yang lainnya
5	Elemen yang satu lebih penting dari pada yang lainnya
7	Satu elemen jelas lebih mutlak penting daripada elemen lainnya
9	Satu elemen mutlak penting daripada elemen lainnya
2, 4, 6, 8	Nilai-nilai antara dua nilai pertimbangan-pertimbangan yang berdekatan
Kebalikan	Jika aktivitas i mendapat suatu angka dibandingkan aktivitas j , maka j memiliki nilai kebalikannya dibandingkan dengan i

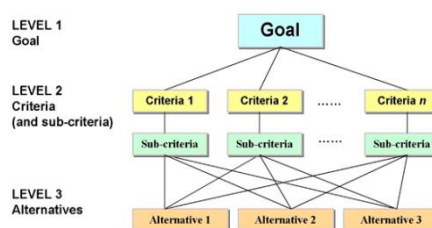
Sumber: [30]

3. *Synthesis of Priority* (menetapkan prioritas)

Penetapan prioritas unsur-unsur standar bisa disangka sebagai bobot elemen untuk menggapai tujuan pengambilan keputusan. Menurut pendapat para ahli dan pengambil keputusan, prioritas ditetapkan secara langsung (dialog) atau tidak langsung (survei kuesioner).

4. *Logical Consistency* (konsistensi logis)

Terdapat dua arti dalam konsistensi yakni tujuan yang seragam dapat dikelompokkan berdasarkan kesamaan serta relevansinya dan susunan ikatan antara objek berlandaskan pada keadaan atau kriteria tertentu. Tahapan dalam prosedur AHP antara lain:



Gambar 1. Struktur Hierarki AHP
Sumber: [31]

1. Identifikasi masalah serta tentukan tujuan yang diinginkan, selanjutnya membuat hierarki agar permasalahan lebih detail dan terukur.
2. Memastikan prioritas elemen
 - a. Membuat perbandingan berpasangan.
 - b. Bilangan yang diisi pada matriks perbandingan berpasangan digunakan untuk menunjukkan kepentingan sesuatu elemen kepada elemen lain.
3. Sintesis

Dalam mendapatkan keseluruhan prioritas dilakukan pertimbangan perbandingan berpasangan dengan proses sintesis. Beberapa hal dalam langkah ini antara lain:

 - a. Nilai dari masing-masing kolom pada matriks dijumlahkan.
 - b. Didapat normalisasi matriks yang diperoleh dari masing-masing bobot dari kolom dibagi jumlah kolom yang sesuai.

- c. Untuk mendapatkan bobot rata-rata dilakukan penjumlahan bobot dari masing-masing baris untuk dibagi dengan jumlah elemen.

4. Menghitung konsistensi

Untuk membuat keputusan, perlu memahami konsistensi yang sesuai, sebab keinginan untuk menghindari keputusan didasarkan pada konsistensi yang buruk. Langkah-langkah dalam hal ini, antara lain:

- a. Masing-masing bobot pada kolom dengan prioritas relatif elemen yang urutannya sama dengan kolom dikalikan sampai seterusnya.
 - b. Masing-masing baris dijumlahkan.
 - c. Lakukan pembagian antara hasil baris yang dijumlahkan dengan elemen prioritas relatif yang sesuai.
 - d. λ maks merupakan hasil dari penjumlahan antara elemen prioritas dengan banyaknya elemen.
5. Rumus Menghitung Indeks Konsistensi (CI):

$$CI = \frac{(\lambda \text{ maks} - n)}{n - 1}$$

Keterangan: n = kuantitatif elemen

6. Menghitung *Consistency Rasio* (CR) sebagai berikut:

$$CR = \frac{CI}{IR}$$

Keterangan:

CR = *Consistency Ratio*

CI = *Consistency Index*

IR = *Index Random Consistency*

7. Mencari konsistensi hierarki

Nilai data *judgment* harus diperbaiki jika nilainya melebihi 10%. Hasil penghitungan dikatakan absolut apabila *Consistency Ratio* (CI/IR) di bawah atau sama dengan 0,1 [31][30].

METODE PENELITIAN

Terdapat beberapa prinsip yang perlu dipahami untuk menyelesaikan masalah dengan *Analytical Hierarchy Process*, yaitu:

- a. *Decomposisi* (membuat hierarki)

Membuat bagian secara hierarki dengan membagi sebuah permasalahan kompleks menjadi bagian lebih kecil supaya dapat dimengerti dengan mudah.

- b. *Comparative Judgment* (evaluasi kriteria serta alternatif)

Perbandingan berpasangan dilakukan untuk mengevaluasi kriteria serta alternative, skala terbaik untuk mengekspresikan pendapat adalah skala 1 sampai 9.

- c. *Synthesis Of Priority* (menetapkan prioritas)

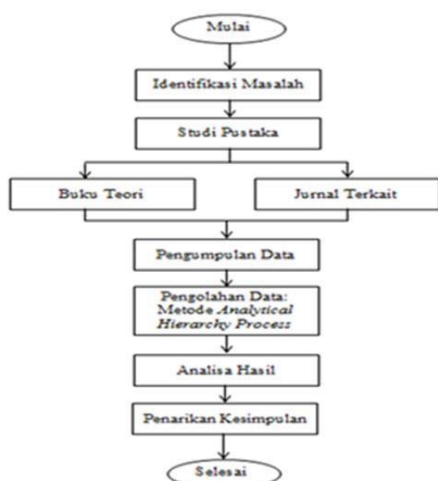
Penetapan prioritas unsur-unsur standar bisa disangka sebagai bobot elemen untuk menggapai tujuan keputusan. Pada penelitian ini, prioritas ditetapkan secara langsung (dialog) atau tidak langsung (survei kuesioner).

- d. *Logical Consistency* (konsistensi logis)

Terdapat dua arti dalam

konsistensi yakni tujuan yang seragam dapat dikelompokkan berdasarkan kesamaan serta relevansinya dan susunan ikatan antara objek berlandaskan pada keadaan ataupun kriteria tertentu [5].

Ada pun langkah-langkah penelitian pada penelitian mengenai pemilihan bahasa pemrograman menggunakan metode AHP, sebagai berikut:



Gambar 2. *Flowchart Tahapan Penelitian*

Analisis Data

Data yang terkumpul dapat dihitung secara manual menggunakan Microsoft Excel. Langkah-langkah pemilihan bahasa pemrograman sebagai berikut:

1. Menyusun struktur hierarki

Saat masalah sudah dapat teridentifikasi kemudian dilakukan penyusunan hierarki solusi atau tujuan yang diinginkan. Tujuan dan kriteria yang perlu dipertimbangkan dalam

pemilihan bahasa pemrograman sebagai berikut:

a. Sederhana/Mudah Dipelajari

Kesederhanaan berarti bahasa pemrograman mudah dipelajari, tata bahasa dan penulisan sintaks mudah diingat. Dalam pembuatan program, seseorang tidak perlu sering melihat panduan dan dokumentasi. Namun, bahasa pemrograman tidak sesederhana itu sehingga tidak memiliki kinerja dan kurang *powerfull* pada tingkat yang lebih tinggi. Bahasa BASIC terlalu sederhana dan menjadi bahasa yang sudah lama ditinggalkan.

b. *Powerfull*

Berarti bahasa pemrograman dapat digunakan untuk melakukan berbagai macam pekerjaan yang berhubungan dengan pemrograman. Bahasa pemrograman mendukung dan mengimplementasi konsep pemrograman modern seperti abstraksi, *thread*, berorientasi objek, pemrograman grafik dua dan tiga dimensi.

c. Populer

Artinya bahasa tersebut memiliki banyak pengguna. Hal ini mempengaruhi ketersediaan dokumentasi pendukung, grup dan forum pengguna, bahkan mempengaruhi lowongan pekerjaan yang terkait dengan keahlian atau pengalaman dalam bahasa pemrograman.

d. *Portable/Cross-Platform*

Portabilitas berarti bahasa pemrograman memiliki beberapa versi yang dapat dijalankan di komputer yang berbeda dan pada platform sistem operasi yang berbeda. Bahasa pemrograman yang portabel memungkinkan membuat program yang portabel yaitu program yang dapat dijalankan atau dikompilasi pada sistem operasi yang berbeda.

2. Memastikan prioritas elemen

- a. Membuat perbandingan berpasangan.
- b. Bilangan yang diisi pada matriks perbandingan berpasangan digunakan untuk menunjukkan kepentingan sesuatu elemen kepada elemen lain.

3. Sintesis

Dalam mendapatkan keseluruhan prioritas dilakukan pertimbangan perbandingan berpasangan dengan proses sintesis. Beberapa hal dalam langkah ini antara lain:

- a. Nilai dari masing-masing kolom pada matriks dijumlahkan.
- b. Didapat normalisasi matriks yang diperoleh dari masing-masing bobot dari kolom dibagi jumlah kolom yang sesuai.
- c. Untuk mendapatkan bobot rata-rata dilakukan penjumlahan bobot dari masing-masing baris untuk dibagi dengan jumlah elemen.

4. Menghitung konsistensi

Untuk membuat keputusan, perlu memahami konsistensi yang sesuai, sebab keinginan untuk menghindari keputusan didasarkan pada konsistensi

yang buruk. Langkah-langkah dalam hal ini, antara lain:

- a. Masing-masing bobot pada kolom dengan prioritas relatif elemen yang urutannya sama dengan kolom dikalikan sampai seterusnya.
- b. Masing-masing baris dijumlahkan.
- c. Lakukan pembagian antara hasil baris yang dijumlahkan dengan elemen prioritas relatif yang sesuai.
- d. λ maks merupakan hasil dari penjumlahan antara elemen prioritas dengan banyaknya elemen.
5. Menghitung Indeks Konsistensi
6. Menghitung *Consistency Ratio*
7. Mencari konsistensi hierarki

Nilai data *judgment* harus diperbaiki jika nilainya melebihi 10%. Hasil penghitungan dikatakan absolut apabila *Consistency Ratio* (CI/IR) di bawah atau sama dengan 0,1 [31][30].

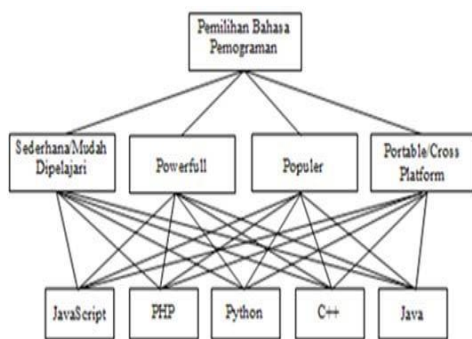
Tabel 2. Indeks Random Konsistensi

Ukuran Matriks	Nilai
1,2	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
11	1,51
12	1,48
13	1,56
14	1,57
15	1,59

Sumber: [30]

MODEL PENGEMBANGAN

Menurut kriteria serta alternatif yang sudah diambil maka dibuatlah suatu hierarki untuk mempermudah dalam pemrosesan data. Struktur hierarki dalam pemilihan bahasa pemrograman untuk mendukung pembuatan web sebagai berikut :



Gambar 3. Struktur Hierarki Pemilihan Bahasa Pemrograman.

HASIL

Hasil dari penelitian ini didapatkan setelah melalui proses pengumpulan data yang dilakukan oleh peneliti untuk mendapatkan data yang dibutuhkan. Dalam penelitian ini data diperoleh dari kuesioner yang disebar untuk mempermudah pengisian dan pengolahan data. Kuesioner yang dibuat berdasarkan beberapa kriteria dan alternatif yang telah ditentukan dalam pemilihan bahasa pemrograman untuk mendukung pembuatan web. Adapun kriteria dan alternatif yang digunakan dalam penelitian ini adalah sebagai berikut:

Tabel 3. Kriteria dan Alternatif

Kriteria	Sederhana/Mudah Dipelajari
	Powerfull
	Populer
	Portable/Cross Platform
Alternatif	JavaScript
	PHP
	Python
	C++
	Java

1. Perhitungan Perbandingan Kriteria

Hasil data responden untuk perbandingan kriteria dari kuesioner yang diolah menghasilkan tabel Data Responden Dari Perbandingan Kriteria

Hasil *geomean* setiap unsur kriteria diubah ke dalam tabel perbandingan matriks dengan sisi diagonal diberikan nilai 1.00 yang berarti netral atau tidak ada yang lebih penting di antara keduanya. Nilai pada sisi atas dari sisi diagonal menyesuaikan nilai *geomean* dengan kriteria yang telah dipasangkan, sedangkan untuk sisi bawah dari sisi diagonal merupakan hasil dari 1 dibagi dengan nilai *geomean* setiap kriteria berpasangan, tabel nilainya sebagai berikut:

Tabel 4. Perbandingan Matriks Kriteria

PERBANDINGAN MATRIKS				
	A	B	C	D
A	1.00	1.75	2.59	1.49
B	0.57	1.00	2.91	1.42
C	0.37	0.34	1.00	1.42
D	0.67	0.70	0.70	1.00
S.O.R	2.62	3.80	7.30	5.34

Keterangan:

A: Sederhana/Mudah Dipelajari

B: Powerfull

C: Populer

D: Portable/Cross Platform

Setelah didapat total jumlah atau S.O.R setiap kriteria selanjutnya hitung bobot relatif yang dinormalisasikan dengan cara nilai kriteria setiap kolom dibagi dengan total jumlah yang didapat. Hasilnya dapat diperoleh pada tabel sebagai berikut:

Tabel 5. Normalisasi Matriks Kriteria

	Matriks				Priority vector	Matriks awal x priority vector	Consistency vector
	A	B	C	D			
A	0.38	0.46	0.37	0.28	0.37	1.57	4.22
B	0.22	0.26	0.40	0.27	0.29	1.57	5.48
C	0.14	0.09	0.14	0.27	0.16	1.57	9.88
D	0.26	0.18	0.10	0.19	0.18	1.57	8.67
S.O.R	1.00	1.00	1.00	1.00	1.00		

Selanjutnya nilai *priority vector* dikalikan dengan total jumlah pada matriks kriteria sesuai barisnya, kemudian hasil dari perhitungan di atas untuk semua nilai dari setiap baris dijumlahkan. Nilai perhitungan tersebut merupakan *principal eigen value* maksimum (λ maks).

λ maks =

$$(2.62 \times 0.37) + (3.80 \times 0.29) + (7.30 \times 0.16) + (5.34 \times 0.18) = 4.19$$

Selanjutnya adalah mencari nilai *Consistency Index* (CI) dengan $n=4$ (sebanyak jumlah kriteria) sesuai rumus seperti berikut:

$$CI = \frac{(\lambda_{\text{maks}} - n)}{(n-1)} = \frac{(4.19 - 4)}{(4-1)} = 0.06$$

Karena matriks berordo 4 berdasarkan nilai n di atas maka indeks random konsistensi (IR) adalah 0,90. Selanjutnya perhitungan *Consistency Ratio* (CR) sebagai berikut:

$$CR = \frac{CI}{IR} = \frac{0.06}{0.90} = 0.07 \text{ atau } 7\%$$

Preferensi responden konsisten jika hasil $CR < 0,1$. Selanjutnya melakukan pengkuadratan matriks dengan cara menyelisihkan hasil pengkuadratan pertama dan kedua, jika selisih *eigen vector* 0 maka bobot kriteria dapat dijadikan sebagai peringkat untuk menentukan prioritas.

Tabel 6. Pengkuadratan Matriks Pertama

	A	B	C	D	S.O.C	1 st Eigen Vector
A	4.00	5.47	11.51	9.30	30.28	0.37914
B	3.18	4.00	8.35	7.84	23.37	0.29265
C	1.90	2.34	4.00	3.89	12.13	0.15184
D	2.01	2.82	5.25	4.00	14.09	0.17637
Total					79.87	1.00

Tabel 7. Pengkuadratan Matriks Kedua

	A	B	C	D	S.O.C	2 nd Eigen Vector
A	73.93	96.93	186.65	162.06	519.56	0.3756
B	57.05	75.07	144.65	124.80	401.56	0.2903
C	30.44	40.07	77.83	67.11	215.45	0.1557
D	35.02	45.85	88.73	77.24	246.85	0.1784
Total					1383.42	1.00

Tabel 8. Peringkat Bobot Kriteria

Kriteria	Peringkat	%
Sederhana/Mudah Dipelajari (A)	1	37.56
Powerfull (B)	2	29.03
Populer (C)	4	15.57
Portable/Cross Platform (D)	3	17.84

Dari hasil perhitungan tabel pengkuadratan kriteria menunjukkan bahwa sederhana/mudah dipelajari merupakan kriteria yang paling penting dalam pemilihan bahasa pemrograman untuk membuat web dengan nilai bobot sebesar 0,37 atau

37%, kedua adalah *powerfull* dengan nilai bobot sebesar 0,29 atau 29%, ketiga adalah *portable/cross platform* dengan nilai bobot sebesar 0,17 atau 17%, dan keempat adalah populer dengan nilai bobot sebesar 0,15 atau 15%.

2. Perhitungan Perbandingan Kriteria Sederhana/Mudah Dipelajari

Hasil *geomean* setiap alternatif diubah ke dalam tabel perbandingan matriks dengan sisi diagonal diberikan nilai 1.00 yang berarti netral atau tidak ada yang lebih penting di antara keduanya. Nilai pada sisi atas dari sisi diagonal menyesuaikan nilai *geomean* dengan alternatif yang telah dipasangkan, sedangkan untuk sisi bawah dari sisi diagonal merupakan hasil dari 1 dibagi dengan nilai *geomean* setiap alternatif berpasangan, tabel nilainya sebagai berikut:

Tabel 9. Kriteria Sederhana/Mudah Dipelajari

PERBANDINGAN MATRIKS					
	A	B	C	D	E
A	1.00	1.13	1.04	2.64	2.72
B	0.89	1.00	1.18	2.71	2.50
C	0.97	0.85	1.00	2.83	1.77
D	0.38	0.37	0.35	1.00	1.37
E	0.37	0.40	0.56	0.73	1.00
S.O.R	3.60	3.74	4.14	9.91	9.36

Keterangan:

A: JavaScript

B: PHP

C: Python

D: C++

E: Java

Setelah didapat total jumlah atau S.O.R setiap alternatif selanjutnya hitung bobot relatif yang dinormalisasikan dengan cara nilai alternatif setiap kolom dibagi dengan total jumlah yang didapat. Hasilnya dapat diperoleh pada tabel sebagai berikut, antara lain:

Tabel 10. Normalisasi Matriks Berpasangan

Matriks					Priority vector	Matriks awal x priority vector	Consistency vector
	A	B	C	D	E		
A	0.28	0.30	0.25	0.27	0.29	0.28	1.40
B	0.25	0.27	0.29	0.27	0.27	0.27	1.40
C	0.27	0.23	0.24	0.29	0.19	0.24	1.40
D	0.11	0.10	0.09	0.10	0.15	0.11	1.40
E	0.10	0.11	0.14	0.07	0.11	0.11	1.40
S.O.R	1.00	1.00	1.00	1.00	1.00	1.00	

Tabel 11. Pengkuadratan Matriks Pertama

	A	B	C	D	E	S.O.C	1st Eigen Vector
A	5.00	5.19	5.87	13.25	13.70	43.02	0.27695
B	4.86	5.00	5.65	12.93	13.21	41.66	0.26821
C	4.40	4.53	5.00	11.79	12.15	37.87	0.24379
D	1.93	2.01	2.31	5.00	5.31	16.56	0.10660
E	1.91	1.96	2.24	5.11	5.00	16.22	0.10445
						155.33	1.00

Tabel 12. Pengkuadratan Matriks Kedua

	A	B	C	D	E	S.O.C	2nd Eigen Vector
A	127.86	132.06	149.34	338.95	347.34	1095.56	0.27714
B	123.72	127.79	144.51	327.95	336.10	1060.07	0.26816
C	112.03	115.71	130.87	296.94	304.28	959.84	0.24281
D	49.38	51.00	57.66	130.91	134.13	423.07	0.10702
E	48.38	49.97	56.50	128.22	131.46	414.54	0.10486
						3953.07	1.00

Tabel 13. Peringkat Bobot Kriteria Sederhana/Mudah Dipelajari

Alternatif	Peringkat	%
JavaScript (A)	1	27.71
PHP (B)	2	26.82
Python (C)	3	24.28
C++ (D)	4	10.70
Java (E)	5	10.49

Dari hasil perhitungan perbandingan alternatif berdasarkan kriteria sederhana/mudah dipelajari menunjukkan bahwa JavaScript merupakan alternatif yang paling baik dalam pemilihan bahasa pemrograman untuk membuat web dengan nilai bobot sebesar 0,27 atau 27%, kedua adalah PHP dengan nilai bobot sebesar 0,26 atau 26%, ketiga adalah Python dengan nilai bobot sebesar 0,24 atau 24%, keempat adalah C++ dengan nilai bobot sebesar 0.10 atau 10%, dan kelima adalah Java dengan nilai bobot sebesar 0,10 atau 10%.

3. Perhitungan Perbandingan Kriteria *Powerfull*

Hasil *geomean* setiap alternatif diubah ke dalam tabel perbandingan matriks dengan sisi diagonal diberikan nilai 1.00 yang berarti netral atau tidak ada yang lebih penting di antara keduanya dengan tabel nilainya sebagai berikut:

Tabel 14. Kriteria *Powerfull*

PERBANDINGAN MATRIKS					
	A	B	C	D	E
A	1.00	2.70	1.55	2.21	1.51
B	0.37	1.00	1.14	1.59	1.18
C	0.64	0.88	1.00	2.06	1.14
D	0.45	0.63	0.49	1.00	1.27
E	0.66	0.85	0.87	0.79	1.00
S.O.R	3.13	6.06	5.05	7.65	6.10

Keterangan:

A: JavaScript

B: PHP

C: Python

D: C++

E: Java

Setelah didapat total jumlah atau S.O.R setiap alternatif selanjutnya hitung bobot relatif yang dinormalisasikan dengan cara nilai alternatif setiap kolom dibagi dengan total jumlah yang didapat. Hasilnya dapat diperoleh pada tabel sebagai berikut, antara lain:

Tabel 15. Normalisasi Matriks Berpasangan

Matriks						Priority vector	Matriks awal x priority vector	Consistency vector
	A	B	C	D	E			
A	0.32	0.45	0.31	0.29	0.25	0.32	1.67	5.18
B	0.12	0.16	0.23	0.21	0.19	0.18	1.67	9.17
C	0.21	0.15	0.20	0.27	0.19	0.20	1.67	8.29
D	0.14	0.10	0.10	0.13	0.21	0.14	1.67	12.19
E	0.21	0.14	0.17	0.10	0.16	0.16	1.67	10.53
S.O.R	1.00	1.00	1.00	1.00	1.00	1.00		

Tabel 16. Pengkuadratan Matriks Pertama

	A	B	C	D	E	S.O.C	1st Eigen Vector
A	5.00	9.45	8.57	13.11	10.79	46.92	0.32626
B	2.97	5.00	4.65	7.27	6.24	26.13	0.18169
C	3.30	5.77	5.00	7.84	6.91	28.83	0.20047
D	2.29	3.99	3.50	5.00	4.52	19.30	0.13422
E	2.56	4.75	4.12	6.19	5.00	22.63	0.15736
						143.80	1.00

Tabel 17. Pengkuadratan Matriks Kedua

	A	B	C	D	E	S.O.C	2nd Eigen Vector
A	139.05	247.47	219.99	333.74	285.35	1225.60	0.32458
B	77.70	138.53	123.12	186.71	159.44	685.49	0.18154
C	85.85	153.03	136.09	206.44	176.22	757.62	0.20065
D	57.91	103.21	91.83	139.47	119.02	511.44	0.13545
E	67.53	120.19	106.94	162.34	138.77	595.76	0.15778
						3775.91	1.00

Tabel 18. Peringkat Bobot Kriteria
Powerfull

Alternatif	Peringkat	%
JavaScript (A)	1	32.46
PHP (B)	3	18.15
Python (C)	2	20.06
C++ (D)	5	13.54
Java (E)	4	15.78

Dari hasil perhitungan perbandingan alternatif berdasarkan kriteria *powerfull* menunjukkan bahwa JavaScript merupakan alternatif yang paling baik dalam pemilihan bahasa pemrograman untuk membuat web dengan nilai bobot sebesar 0,32 atau 32%, kedua adalah Python dengan nilai bobot sebesar 0,20 atau 20%, ketiga adalah PHP dengan nilai bobot sebesar 0,18 atau 18%, keempat adalah Java dengan nilai bobot sebesar 0,15 atau 15%, dan kelima adalah C++ dengan nilai bobot sebesar 0,13 atau 13%.

4. Perhitungan Perbandingan Kriteria *Portable/Cross Platform*

Hasil *geomean* setiap alternatif diubah ke dalam tabel perbandingan matriks dengan sisi diagonal diberikan nilai 1.00 yang berarti netral atau tidak ada yang lebih penting di antara keduanya. Nilai pada sisi atas dari sisi diagonal menyesuaikan nilai *geomean* dengan alternatif yang telah dipasangkan, sedangkan untuk sisi bawah dari sisi diagonal merupakan hasil dari 1 dibagi dengan nilai *geomean* setiap alternatif berpasangan. Dari hasil perhitungan perbandingan alternatif berdasarkan kriteria *portable/cross platform* menunjukkan bahwa JavaScript merupakan alternatif

yang paling baik dalam pemilihan bahasa pemrograman untuk membuat web dengan nilai bobot sebesar 0,42 atau 42%, kedua adalah Python dengan nilai bobot sebesar 0,17 atau 17%, ketiga adalah PHP dengan nilai bobot sebesar 0,16 atau 16%, keempat adalah Java dengan nilai bobot sebesar 0,12 atau 12%, dan kelima adalah C++ dengan nilai bobot sebesar 0,09 atau 9%.

5. Perhitungan Hasil Akhir Kriteria dan Alternatif

Tabel 19. Hasil Kriteria dan Alternatif

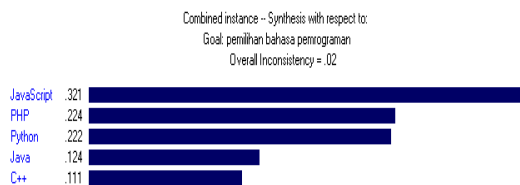
Alternatif	Sederhana/Mudah Dipelajari	Powerfull	Populer	Portable/Cross Platform
JavaScript (A)	27.71%	32.46%	33.70%	42.61%
PHP (B)	26.82%	18.15%	22.44%	16.90%
Python (C)	24.28%	20.06%	23.69%	17.79%
C++ (D)	10.70%	13.54%	9.38%	9.83%
Java (E)	10.49%	15.78%	10.79%	12.84%

Tabel 20. Hasil Akhir Bobot Persentase Alternatif

Alternatif	Score	Peringkat
JavaScript (A)	32.68%	1
PHP (B)	21.85%	2
Python (C)	21.81%	3
C++ (D)	11.17%	5
Java (E)	12.49%	4

Berdasarkan penjelasan diatas maka pemilihan bahasa pemrograman untuk membuat web yang paling banyak digunakan adalah bahasa pemrograman JavaScript dengan total bobot

persentase sebesar 32,68%. Kemudian *PHP* di posisi kedua dengan persentase sebesar 21,85%, ketiga adalah *Python* dengan persentase sebesar 21,81%, keempat adalah *Java* dengan persentase sebesar 12,49%, dan kelima adalah *C++* dengan persentase sebesar 11,17%.



Gambar 4. Grafik Urutan Hasil Akhir Alternatif

Jika dibandingkan dengan hasil pengolahan data menggunakan *Expert Choice* di atas maka hasil perbandingannya sama dengan pengolahan data menggunakan *Ms.Excel* sama berdasarkan kesesuaian urutan seluruh alternatif dan kriteria.

KESIMPULAN

Pemilihan bahasa pemrograman untuk mendukung pembuatan web menggunakan metode *Analytical Hierarchy Process* (AHP) terkait pemilihan bahasa pemrograman untuk mendukung pembuatan web menghasilkan penilaian yang cukup untuk memberikan informasi dan referensi yang dibutuhkan, dapat menjadi bahan pertimbangan yang berguna untuk para pemula atau programmer yang membutuhkannya dalam mendukung proses pembuatan web.

Berdasarkan hasil perhitungan dari lima bahasa pemrograman yang dipilih ada *JavaScript* dengan hasil perhitungan sebesar 32,68% menjadi bahasa pemrograman paling banyak digunakan dan berada di tingkat atas pilihan dari setiap kriteria kemudian bahasa lainnya ada *PHP* dengan persentase sebesar 21,81%, *Python* dengan persentase sebesar 21,81%, *Java* dengan persentase sebesar 12,49%, serta *C++* dengan persentase sebesar 11,17%. Dengan kriteria bahasa pemrograman sederhana/mudah dipelajari menjadi kriteria paling banyak dipilih dengan persentase tertinggi dari kriteria *powerfull*, populer, dan *portable/cross platform*.

DAFTAR PUSTAKA

- [1] R. Dewi, P. A. Janitra, and N. Aristi, "Pemanfaatan Internet Sebagai Sumber Informasi Kesehatan Bagi Masyarakat," *Media Karya Kesehat.*, vol. 1, no. 2, pp. 162–172, 2018.
- [2] Admin, "Statistik nama domain tahun 2020.xlsx," *Satu Data Indonesia*, 2021. [Online]. Available: <https://data.kominfo.go.id/dataset/bd6e9cf7-f5b4-4b61-abb3-a93994c982f9/resource/8586f3e3-d92f-4e7a-acde-a0d4a975741a/download/statistik-nama-domain->

- tahun-2020.xlsx. [Accessed: 02-Jun-1BC].
- [3] P. S. Hasugian, “Perancangan Website Sebagai Media Promosi Dan Informasi,” *J. Inform. Pelita Nusan.*, vol. 3, no. 1, pp. 82–86, 2018.
- [4] O. Ezenwoye, “What language? - The choice of an introductory programming language,” *Proc. - Front. Educ. Conf. FIE*, vol. 2018–Octob, pp. 1–8, 2019.
- [5] A. Priyanto, “10 Bahasa Pemrograman Populer Di Indonesia,” *Code Politan*, 2019. [Online]. Available: <https://www.codepolitian.com/10-bahasa-pemrograman-populer-di-indonesia/>. [Accessed: 02-Jun-1BC].
- [6] D. R. Prehanto, C. Mashuri, and A. D. Indriyanti, *Buku Ajar Model Sistem Pendukung Keputusan Dengan AHP dan IPMS*, 1st ed. Surabaya: Scorpio Media Pustaka, 2020.
- [7] L. Yulianti, “Implementasi Metode AHP dalam Menentukan Penerima Beasiswa Komite Sekolah,” vol. 5, no. April, pp. 746–755, 2021.
- [8] N. Andriyani and A. Hafiz, “Perbandingan Metode AHP dan Topsis dalam Penentuan Siswa Berprestasi,” *Semin. Nas. Teknol. Dan Bisnis 2018*, pp. 362–371, 2018.
- [9] S. Nurhidayah, M. N. Fauzan, and W. I. Rahayu, *Impelementasi Metode Analytical Hierarchy Process (AHP) dengan PHP*, 1st ed. Bandung: Kreatif Industri Nusantara, 2020.
- [10] Admin, “Pembahasan Lengkap Teori Analytical Hierarchy Process (AHP) menurut Para Ahli dan Contoh Tesis Analytical Hierarchy Process (AHP),” *idtesis.com*, 2018. [Online]. Available: <https://idtesis.com/pembahasan-lengkap-teori-analytical-hierarchy-process-ahp-menurut-para-ahli-dan-contoh-tesis-analytical-hierarchy-process-ahp/>. [Accessed: 02-Jun-1BC].