

Penerapan AHP dan Normalisasi SAW sebagai Pendukung Promosi Ide pada *Mobile IMS*

M. Fikri Rizki Romadhoni¹, Aditya Prapanca²

^{1,2}Jurusan Teknik Informatika Fakultas Teknik Universitas Negeri Surabaya

¹muhamad.18060@mhs.unesa.ac.id

²adityaprapanca@unesa.ac.id

Abstrak— Membuat sebuah ide merupakan kegiatan mencari solusi, mengolah pikiran, menganalisis, menggali dan mengevaluasi pengetahuan. Teknologi informasi belakangan ini sudah mencakup banyak bidang termasuk sistem manajemen ide. teknologi di bidang manajemen ide bertujuan utama untuk menghasilkan ide yang lebih baik dengan melibatkan banyak orang untuk saling mengevaluasi. Pada praktik pengelolaan ide yang masih menggunakan cara tradisional, terdapat banyak hambatan yang dipengaruhi oleh banyak faktor yaitu faktor internal dari individu itu sendiri maupun faktor eksternal. Kehadiran teknologi menjadikan interaksi pada kasus manajemen ide lebih interaktif sehingga sekelompok orang dapat berdiskusi untuk mengevaluasi ide yang berawal dari gagasan satu orang menjadi buah pikiran banyak orang. Interaksi ini merupakan salah satu yang menjadi acuan bahwasanya kualitas ide atau gagasan tersebut patut dipertimbangkan. Dalam penelitian ini dikembangkan sistem manajemen ide yang menerapkan metode pendukung keputusan dengan mengukur secara matematis terhadap interaksi dan poin-poin penting dalam sebuah ide. Metode *Analytical Hierarchy Process* dan metode *Simple Additive Weighting* digunakan sebagai acuan pemilihan ide terbaik dalam sebuah kelompok berdasarkan tingkat kepentingan kriteria yang diharapkan. Dalam pengujian iterasi, perhitungan metode AHP dan SAW dapat diimplementasikan pada program aplikasi. Iterasi program yang dilakukan dapat menghitung nilai dari data kuantitatif. Dalam penerapan metode, terdapat beberapa kekurangan yang menyebabkan sistem menjadi sulit dipergunakan. Modifikasi metode AHP menghasilkan peluang konsistensi *input* yang sebelumnya hanya 17,484% meningkat menjadi 99,98% dan mudah untuk dipergunakan. Dari modifikasi yang dilakukan akan menghindarkan pengguna untuk melakukan *input* yang tidak konsisten. Penggunaan kombinasi metode AHP dan SAW akan meningkatkan performa perhitungan berdasarkan waktu eksekusi dan penggunaan sumber daya perangkat.

Kata Kunci— *Analytical Hierarchy Process, Simple Additive Weighting, Idea Management System, Sistem Pendukung Keputusan, Mobile Application.*

I. PENDAHULUAN

Pada era modernisasi dan digitalisasi saat ini, berdampak pada kemudahan dalam menyebarkan sebuah informasi maupun pendapat. Dalam sebuah perusahaan maupun institusi, sebuah ide ataupun inovasi merupakan hal yang sangat diperlukan agar tetap dapat bertahan ketika dihadapkan dengan permasalahan maupun tantangan di masa mendatang. Program manajemen ide yang beroperasi dengan sejumlah besar ide dapat mendukung perusahaan dalam menjangkau pasar lebih

cepat serta memenuhi tuntutan preferensi konsumen atau iklim politik yang berubah [6]. Namun cara tradisional yang umum digunakan seperti menyediakan *box* surat penyampaian aspirasi maupun inovasi dianggap kurang efektif mengingat zaman yang sudah beralih ke fase digitalisasi. Selain itu, faktor yang menyebabkan kurang efektifnya metode tersebut adalah kemungkinan terlalu banyaknya jumlah ide yang diberikan sehingga mempersulit dalam pengelompokan serta penentuan kualitas ide terbaik berdasarkan solusi yang diberikan dan kriteria lainnya. Saat mengevaluasi ide, penilai dapat dengan cepat mengalami kelebihan beban kognitif yang dapat mengakibatkan kinerja seleksi yang buruk [2].

Beberapa sistem informasi telah dikembangkan belakangan terakhir untuk mendapatkan lebih banyak dari kolaborasi, manajemen pengetahuan, dan ide [9]. Banyak perusahaan berlomba-lomba dalam meningkatkan kemampuannya dalam mencetuskan ide-ide kreatif serta inovasi yang membuat perusahaan tersebut mampu menjawab segala tantangan yang mereka hadapi. Penggunaan *idea management* berbasis web telah menjadi bagian dari budaya inovasi yang digunakan oleh banyak organisasi terkenal di dunia di berbagai sektor, seperti Fujitsu, Electrolux, Heineken, NASA, Panasonic, Sony, Volvo [8]. Namun kemudahan memberikan ide harus diikuti dengan peningkatan dalam menyeleksi setiap ide yang ada guna hanya ide-ide terbaik dan berkualitas sajalah yang berpotensi untuk dipromosikan maupun diterapkan oleh seluruh anggota.

Berdasarkan penjabaran terkait masalah dalam mengakomodir sejumlah ide dan peralihan metode tradisional menjadi lebih modern dengan digitalisasi, maka dibutuhkan sistem yang mampu menampung setiap ide yang dicetuskan dari setiap anggota perusahaan, mengakomodirnya agar dapat berkolaborasi dengan anggota yang lain, serta berinteraksi untuk memperbaiki ide yang ada menjadi lebih baik. *Idea Management System (IMS)* merupakan alat sistematis yang dapat dikelola untuk menghasilkan dan mengevaluasi ide [9]. Dalam proses menghasilkan sebuah ide, terdapat interaksi kolaborasi yang saling mengevaluasi dan menghasilkan data yang dapat diolah. Dalam menyeleksi ide, seseorang terkadang ragu-ragu dan mengalami kesulitan menentukan preferensinya di antara ide-ide yang dianggap setara. Dalam hal ini, seseorang cenderung menyukai ide-ide yang sudah dipilih oleh orang lain sebelumnya [4]. Oleh karena itu, teknik yang dapat digunakan untuk menyeleksi sekumpulan ide guna pengadaan promosi ide adalah konsep dari MADM (*Multiple Attribute Decision Making*) atau bisa disebut sistem pendukung keputusan (SPK)

yang menentukan alternatif ide yang paling sesuai dengan kriteria yang diinginkan.

Sistem bernama IdeaBox merupakan salah satu *Idea Management System* dengan antar muka yang dibangun menggunakan *Framework React JavaScript*. Untuk pengembangan berbasis *mobile* dapat mempergunakan *Framework React Native* yang mendukung *multiplatform*. *React Native* merupakan kerangka kerja *javascript* yang digunakan untuk membangun aplikasi *mobile* Android maupun iOS. *React Native* ini memiliki dasar dari *React* dan *library javascript* dalam membangun antarmuka. *React Native* ini ditulis dengan campuran *javascript* dan *JSX*, lalu *React Native* ini juga memaparkan antarmuka *javascript* untuk *platform* API [11]. Pengembangan aplikasi *mobile* pada sistem IdeaBox bertujuan untuk memenuhi kebutuhan fungsionalitas di masa mendatang serta sebagai sarana mendapatkan *user engagement*. Di sisi lain, aplikasi *Web Based* dan *Mobile Based* pada sistem IdeaBox memiliki arsitektur yang sama yaitu diatas *NodeJs* sehingga tidak dibutuhkan biaya tambahan yang terlalu besar. Mengingat atas teori bahwa ide dapat muncul sewaktu-waktu dan harus segera dituangkan, maka pengembangan sistem IdeaBox sangat cocok dikembangkan menjadi aplikasi *mobile*. Arsitektur *hybrid* seperti *React Native* cenderung lebih mudah dipahami dan dikuasai banyak orang. *React Native* kini telah hadir dengan konsep *hooks* yang artinya berbasis *functional component*. *React Native* terbukti bermanfaat bagi pengembang atau tim yang memiliki anggaran terbatas karena aplikasi dapat mendukung beberapa *platform* utama dengan basis kode yang sama. *React* juga telah mendapatkan popularitas di kalangan pengembang perangkat lunak karena fleksibilitas, penggunaan kembali, dan skalabilitasnya [7].

Dalam penelitian ini akan dibuat sebuah aplikasi *mobile* berdasarkan sistem IdeaBox dengan data yang tersedia sejak awal tahun 2022. Setiap ide atau inovasi akan memiliki beberapa atribut induktif berbasis data seperti jumlah kolaborator, nilai *lean-canvas*, jumlah interaksi komentar dan *like*, serta atribut lainnya yang dikelompokkan berdasarkan lingkup institusi tertentu dan tidak mengikutsertakan data yang berasal dari institusi lain. Penelitian terdahulu terkait penggunaan kombinasi metode AHP dan SAW yang dilakukan oleh Daniawan (2018)[3] dengan judul “*Evaluation of Lecturer Teaching Performance Using AHP and SAW Methods*” menghasilkan hasil yang lebih efektif dan efisien serta tingkat konsistensi 90.39% dari 10 indikator untuk evaluasi kinerja dosen. *Analytic Hierarchy Process* (AHP) adalah metode pengambilan keputusan dengan menggunakan perbandingan berpasangan dari setiap kriteria. Teori ini pertama kali diperkenalkan oleh Thomas Saaty [1]. Dalam penggunaan metode AHP, perlu dilakukan dekomposisi masalah dengan mengidentifikasi kriteria dan subkriteria yang akan digunakan [13]. Dalam prakteknya metode ini tidak hanya melakukan iterasi tingkat prioritas, namun juga memperhitungkan tingkat konsistensi penentuan tingkat kepentingan yang diberikan. Metode AHP akan mempertimbangkan tingkat kepentingan antar kriteria yang ada pada sistem penentuan serta nilai konsistensinya. Untuk itu, tingkat kepentingan kriteria berdasarkan atribut ide harus ditentukan. Di sisi lain,

karakteristik ini yang menjadikan metode AHP memiliki masalah jika tidak dilakukan oleh orang yang benar-benar tepat dan memiliki pemahaman tinggi terhadap aspek yang dibandingkan. Dengan teknik skala terhitung, persepsi tingkat kepentingan kriteria akan dihitung berdasarkan perbandingan yang dilakukan secara bersamaan karena akan mengurangi potensi tidak konsistennya persepsi tingkat kepentingan kriteria yang diberikan. Pada penelitian “*Penerapan Metode Analytical Hierarchy Process dalam Sistem Pendukung Keputusan Penentuan Mahasiswa Berprestasi*” yang dilakukan oleh Munthafa (2017)[10] menyatakan bahwa AHP dapat dijadikan sebagai alat bantu dalam menentukan mahasiswa berprestasi dengan adanya data kuantitatif serta adanya tingkat validitas konsistensi hierarki. Selain itu, penelitian ini juga mengkombinasikan dengan metode *simple additive weighting* (SAW) atau yang disebut perhitungan bobot dari hasil persentase yang diperoleh dari metode AHP untuk memperhitungkan dengan setiap nilai kriteria yang terdapat pada setiap alternatif ide atau inovasi. Metode ini dipilih karena memiliki kelebihan yaitu prosesnya yang lebih singkat untuk menentukan alternatif ide untuk dipromosikan. Pada penelitian berjudul “*Metode Simple Additive Weighting Pada Penentuan Penerimaan Karyawan*” yang dilakukan oleh Ermin (2019)[5] menyatakan bahwa penggunaan metode SAW dapat membantu dalam pengambilan keputusan di suatu kasus. Proses perhitungan dengan metode SAW berdasarkan hasil dengan nilai terbesar yang akan terpilih sebagai alternatif terbaik. Metode SAW merupakan salah satu pendukung keputusan yang ringkas dengan menambahkan bobot dari setiap kriteria. Metode ini membutuhkan proses normalisasi matriks keputusan (X) ke skala yang sebanding dengan semua peringkat alternatif yang ada [12]. Dalam hal ini normalisasi dibagi menjadi dua berdasarkan sifatnya yaitu *benefit* (keuntungan) dan *cost* (biaya)[14]. Maka dari itu, data yang terdapat pada setiap kriteria dari setiap alternatif dinormalisasikan berdasarkan sifatnya yang merupakan *benefit* atau *cost*. Perpaduan nilai bobot dengan nilai normalisasi akan menghasilkan urutan ide sebagai alternatif yang dapat dijadikan acuan untuk menentukan ide dengan kriteria terbaik. Dengan dibuatnya sistem manajemen promosi ide terbaik pada penelitian ini, penulis berharap dapat membantu pihak perusahaan, institusi maupun organisasi dalam meningkatkan efektifitas pencetus ide serta fleksibilitas dalam menghadapi tantangan yang ada.

II. METODE PENELITIAN

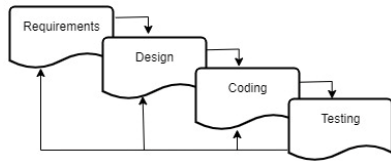
A. Metode Pengumpulan Data

Metode pengumpulan data yang dilakukan pada penelitian ini adalah studi pustaka terhadap sejumlah literatur dan observasi data pengembangan pada sistem IdeaBox yang sudah ada. Data yang berasal dari server diambil menggunakan *http request* dengan *React Native axios library* yang menghasilkan *JSON response* menjadi data penelitian.

B. Metode Pengembangan Perangkat Lunak

Pada penelitian ini pengembangan aplikasi akan diimplementasikan pada sisi *front-end mobile* menggunakan

metode *Waterfall*. Metode ini merupakan hal yang menggambarkan pendekatan *step by step* secara sistematis dalam mengembangkan sebuah perangkat lunak. Tahapan awal dimulai dengan mendefinisikan spesifikasi kebutuhan lalu dilanjutkan dengan perancangan desain sistem, pembuatan kode program, hingga pengetesan perangkat lunak [15]. Tahapan dari alur penelitian dapat diilustrasikan seperti gambar 1 berikut.



Gbr. 1 Metode Waterfall

Tahapan terkait metode *waterfall* pada penelitian ini yaitu:

1) *Requirements* (Kebutuhan sistem): Proses ini merupakan analisis kebutuhan awal berupa penentuan macam-macam data dalam merancang sebuah sistem serta hubungan data yang dapat mempertimbangkan fitur apa saja yang harus dibuat.

2) *Design* (Desain): Pada tahap ini dilakukan peninjauan hasil dari analisis kebutuhan kedalam bentuk desain aplikasi yang tujuannya agar sistem dapat dipergunakan oleh pengguna dan mempermudah pemahaman dalam menjalankan aplikasi/program sesuai dengan kebutuhan yang diharapkan.

3) *Coding* (Pembuatan Kode): *Coding* merupakan tahapan untuk mengimplementasikan desain yang sudah dibuat serta implementasi algoritma dan logika ke dalam bentuk bahasa pemrograman yang dapat menghasilkan sebuah aplikasi/program yang dapat dipergunakan. Dalam penelitian ini aplikasi yang dihasilkan berupa aplikasi mobile.

4) *Testing* (Pengujian): Tahap akhir merupakan pengujian aplikasi yang dilakukan untuk menguji ketepatan pengolahan data berdasarkan algoritma yang diterapkan menggunakan perhitungan manual serta menguji hasil kehandalan antara algoritma asli dengan yang sudah dimodifikasi.

Adapun alur penggunaan aplikasi yang menjadi bagian penting dalam tersedianya data pada sistem yaitu produksi ide, interaksi pengguna lain, kolaborasi serta evaluasi. Alur sistem ini akan diimplementasikan kedalam antarmuka yang dibangun menggunakan *Framework React Native* menjadi aplikasi *mobile* dengan fitur-fitur yang merepresentasikan fungsi minimal dari aksi yang dibutuhkan.

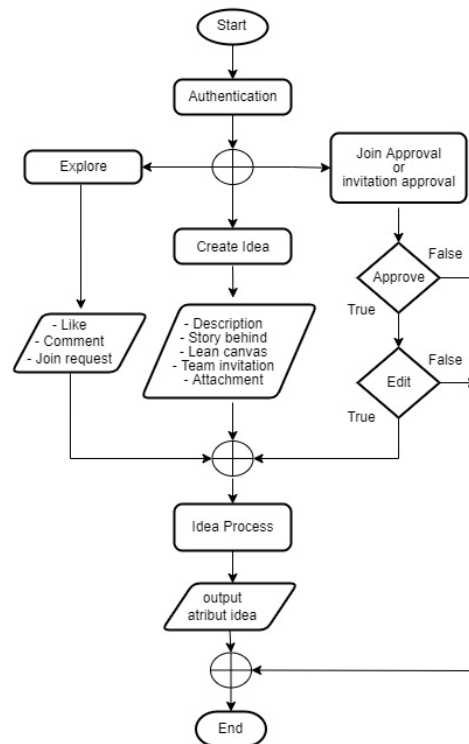
Ilustrasi pada gambar 2 adalah proses penggunaan sistem dalam memproduksi sebuah ide berdasarkan interaksi pengguna yaitu:

- Autentikasi – pengecekan pengguna berdasarkan lingkup email terdaftar. Proses autentikasi akan mengembalikan *user token* yang berguna untuk menyimpan data pribadi serta sesi masuk pengguna dalam beberapa waktu tertentu. Proses ini akan terus berulang untuk me-*refresh token* dengan yang baru.
- *Explore Idea* - dimana semua kumpulan data ide yang sudah tersubmit sebelumnya akan ditampilkan. Pada sesi

ini seluruh pengguna dapat bebas berinteraksi dengan memberikan *like* maupun komentar berdasarkan ketertarikannya. Diperlihatkan profil kreator ide serta dimungkinkan untuk melihat detail ide. Dengan adanya pemikiran yang sama, pengguna lain bisa melakukan *join request* untuk menjadi tim kolaborasi dan dapat saling mengevaluasi dalam ide tersebut.

- *Create Idea* - memungkinkan pengguna untuk memproduksi sebuah ide baru yang terpikirkan dengan beberapa tahapan seperti memberikan informasi deskripsi, alasan ide (*story behind*), poin-poin pada *lean canvas*, lampiran tambahan sebagai pendukung, serta *user invitation* guna mengajak secara langsung pengguna lain untuk berkolaborasi.

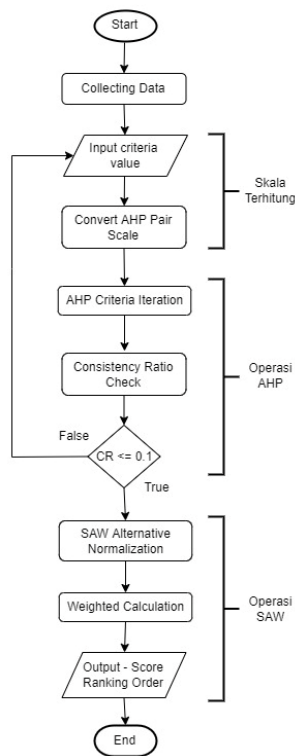
Approval - terbagi menjadi 2 yaitu *join approval* dan *invitations approval*. *Join approval* adalah pilihan untuk menerima atau menolak permintaan pengguna lain untuk *join* dengan ide terkait sedangkan *invitations approval* adalah pilihan untuk menerima atau menolak *invitation* dari suatu ide yang di-*submit* oleh orang lain.



Gbr. 2 Alur Penggunaan Aplikasi

Selanjutnya adalah proses yang mengimplementasikan algoritma penentuan pendukung keputusan untuk menyeleksi dan mengurutkan sejumlah ide dengan kriteria yang diinginkan oleh seorang *innovation manager* saat pengadaan promosi ide. Alur tersebut diilustrasikan dengan gambar berikut.

Pada *flowchart* yang diilustrasikan pada gambar 3 menjelaskan beberapa tahapan implementasi algoritma yang digunakan ke dalam sistem untuk menentukan skor ide berdasarkan preferensi seorang *innovation manager*.



Gbr. 3 Alur Proses Pendukung Keputusan Ide Terbaik

Penjelasan dari alur tersebut yaitu:

- Proses diawali dengan mengumpulkan data-data atribut beserta detail ide yang akan menjadi alternatif untuk diakumulasikan berdasarkan nilai berbasis data induktif.
- Selanjutnya pada penelitian ini dikembangkan sebuah cara untuk memodifikasi tahapan metode AHP yaitu dengan memberikan langkah tambahan yang menggantikan proses pemberian nilai tingkat kepentingan kriteria secara berpasangan. Proses ini dilakukan dengan cara memberikan nilai kepada setiap kriteria secara langsung dan mempertimbangkan kesejajaran nilai dengan kriteria yang lainnya. Ini bertujuan untuk mengurangi jumlah *input* yang diberikan terlebih ketika ada banyak kriteria yang dibandingkan.
- Proses dilanjutkan dengan mengubah nilai masukan yang diberikan kepada setiap kriteria kedalam skala perbandingan pada metode AHP yaitu perbandingan berpasangan berupa angka 1 sampai 9 beserta angka kebalikannya.
- Setelah nilai perbandingan didapat, sistem akan melakukan iterasi AHP dan menghasilkan nilai *Priority vector* untuk setiap kriteria yang ada.
- Sebenarnya proses iterasi AHP selesai disini, namun penting untuk dilakukan proses perhitungan tingkat konsistensi perbandingan kriteria berdasarkan nilai *consistency ratio*. Untuk itu, dilakukan proses tambahan yaitu mencari nilai *principle eigen value* dan *consistency index* yang menghasilkan nilai *consistency ratio*.

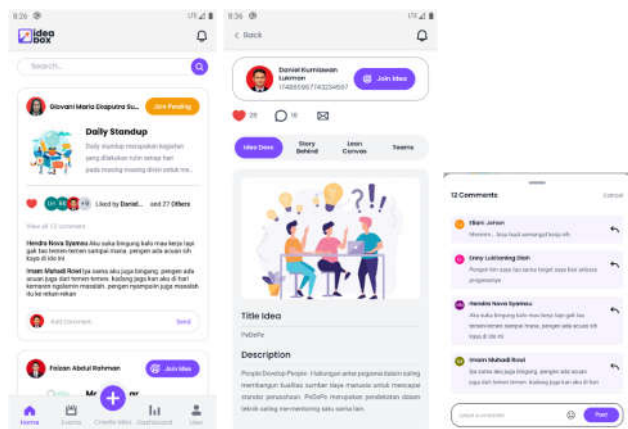
- Setelah nilai *consistency ratio* didapat, dilakukan pengecekan apakah nilai CR diatas 10%. Konsistensi dikatakan cukup apabila nilai CR tidak melebihi 0.1 atau 10%. Apabila syarat rasio konsistensi terpenuhi maka proses dapat dilanjutkan.
- Proses selanjutnya adalah mengubah nilai-nilai atribut alternatif kedalam satuan yang sama pada tahapan normalisasi SAW. Pada tahap ini sistem akan menormalisasikan data berdasarkan sifat kriteria yang telah ditentukan apakah *cost* atau *benefit*.
- Setelah itu pada metode SAW dilanjutkan pada tahapan dimana setiap nilai atribut alternatif dikalikan dengan nilai pembobotan lalu diakumulasikan menjadi nilai skor untuk masing-masing alternatif.
- Di akhir perhitungan, dilakukan *sorting* berdasarkan nilai skor tertinggi dan akan menjadi acuan untuk menentukan ide yang paling sesuai dengan ketentuan kriteria yang diinginkan.

Alur program yang diilustrasikan pada gambar 2 dan 3 perlu diperhatikan bahwasanya hasil akhir yang dihasilkan bukan acuan mutlak dikarenakan sistem yang menerapkan metode hanyalah alat pendukung. Untuk itu, pada hasil perancangan disediakan skenario untuk melihat detail ide berdasarkan urutan skor tertinggi untuk dilakukan tinjauan manual.

III. HASIL DAN PEMBAHASAN

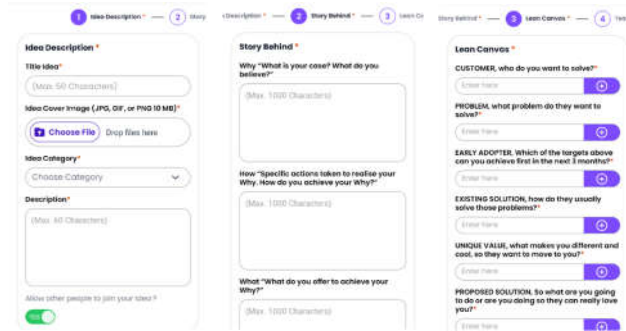
A. Implementasi Program Antarmuka

Penelitian ini menggunakan platform *mobile* untuk penerapan program. Bahasa pemrograman Javascript digunakan untuk mengembangkan aplikasi dengan *Framework React Native* untuk menghasilkan aplikasi *hybrid*. Implementasi aplikasi *mobile* pada sistem IdeaBox bertujuan untuk memenuhi kebutuhan fungsionalitas di masa mendatang, mendapatkan interaksi yang lebih alami, kemudahan akses, serta sebagai sarana mendapatkan *user engagement*. Di sisi lain, aplikasi *Web Based* dan *Mobile Based* pada sistem IdeaBox memiliki arsitektur yang sama yaitu diatas *NodeJs* sehingga tidak dibutuhkan biaya tambahan yang terlalu besar.

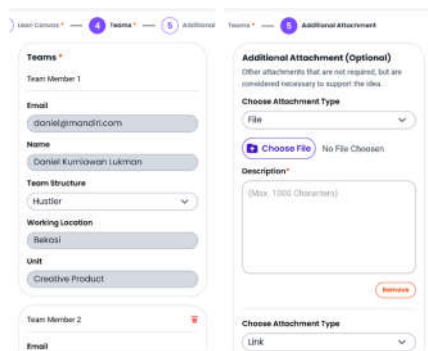


Gbr. 4 Tampilan Explore, Idea detail, dan Comment

Pada gambar 4 merupakan tampilan utama aplikasi yang dapat dipergunakan pengguna lain untuk mengeksplorasi sekumpulan ide-ide yang sudah dibuat. Pengguna dapat berinteraksi dengan memberikan *like* maupun memberikan komentar. Gambar 4 bagian tengah merupakan halaman yang memperlihatkan detail ide beserta tombol aksi untuk melakukan *join request*. Aksi *join request* akan memberikan status *pending* seperti yang ada pada gambar 4 sebelah kiri hingga kreator ide menerima permintaan tersebut. Untuk melihat komentar keseluruhan dapat membuka opsi lihat semua komentar seperti yang dicontohkan pada gambar 4 sebelah kanan. Sedangkan untuk membuka halaman guna membuat sebuah ide baru dapat diakses dengan tombol *Plus* pada *Tab Bar* seperti yang terlihat pada gambar 4 sebelah kiri. Pada gambar 5 dan gambar 6 merupakan halaman untuk membuat sebuah ide baru dengan mengisi deskripsi, *story behind*, *lean canvas*, *team invitations*, serta lampiran pendukung.



Gbr. 5 Tampilan Create Idea bagian Idea Description, Story Behind, dan Lean Canvas.



Gbr. 6 Tampilan Create Idea bagian Team Invite dan Additional Attachment

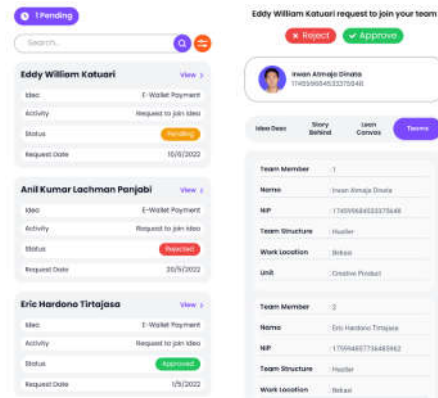
Jika pada langkah *team invite* menambahkan pengguna lain untuk bergabung, maka pengguna terkait akan menerima pemberitahuan seperti pada gambar 7.



Gbr. 7 Notifikasi Invitations

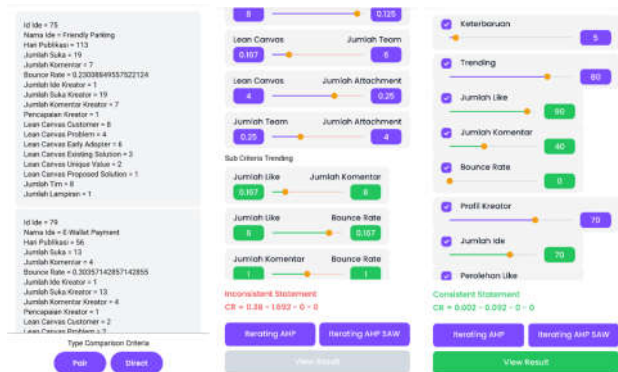
Apabila pengguna merupakan kreator sebuah ide tertentu dan terdapat pengguna lain yang melakukan *join request* seperti

pada gambar 4 sebelah kiri, maka kreator dapat memilih untuk menerima atau menolaknya pada menu *Talent Approval* seperti pada gambar 8.



Gbr. 8 Talent Approval

Untuk keperluan seleksi ide yang dilakukan oleh seorang *innovator manager* dibuat antarmuka yang mengimplementasikan tahapan penentuan tingkat kepentingan suatu kriteria. Pada gambar 9 bagian tengah merupakan bentuk implementasi perbandingan berpasangan sebagaimana langkah awal dalam metode AHP. Namun untuk keperluan penelitian, disediakan pula cara alternatif seperti pada gambar 9 sebelah kanan. Terlihat jelas perbedaan antara keduanya dimana pada cara yang pertama menentukan tingkat antar dua kriteria secara berpasangan sedangkan cara kedua hanya menentukan nilai dari masing-masing kriteria. Cara ini akan mempengaruhi beberapa aspek pada penjelasan iterasi.

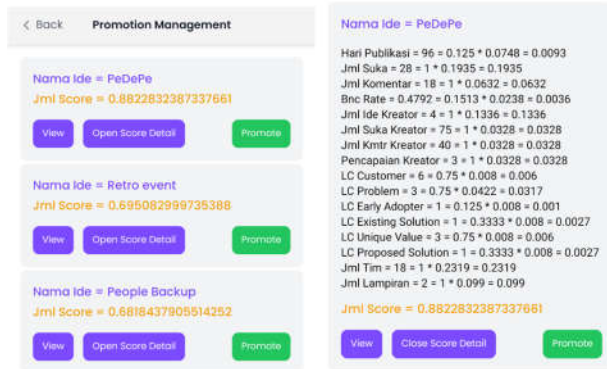


Gbr. 9 Criteria Input

Desain ukuran dan warna *input* mengartikan bahwa warna biru dengan ukuran lebih besar merupakan kriteria utama. Sedangkan untuk warna hijau dengan ukuran lebih kecil merupakan sub kriteria yang dapat dilihat pada gambar 9 bagian tengah dan bagian kanan. Nilai masukkan yang diberikan akan memandu sistem dalam melakukan iterasi dan menentukan skor pada setiap data ide. Setelah sistem selesai melakukan iterasi, pada gambar 10 memperlihatkan bahwa sistem mengembalikan urutan ide dengan skor tertinggi yang dapat dijadikan acuan dalam mempertimbangkan promosi ide.

TABEL II
NILAI RANDOM CONSISTENCY INDEX

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
RI	0	0	0,52	0,89	1,11	1,25	1,35	1,4	1,45	1,49	1,51	1,54	1,56	1,57	1,58



Gbr. 10 Hasil Skoring

B. Implementasi Program Algoritma

Penelitian ini mengimplementasikan tahapan-tahapan algoritma dari metode AHP dan SAW yang menjalankan sebuah proses iterasi dalam bentuk program. Dalam menentukan jumlah perbandingan berpasangan, terdapat persamaan sebagai berikut dimana simbol (*P*) merupakan banyaknya perbandingan berpasangan dan simbol (*n*) merupakan banyaknya kriteria..

$$P = n^2 - n / 2 \tag{1}$$

Nilai skala yang umum digunakan pada metode AHP seperti pada tabel 1.

TABEL I
SKALA KEPENTINGAN AHP

Nilai	Keterangan
1	Sama penting
3	Lebih penting sedikit
5	Lebih penting secara kuat
7	Lebih penting sangat kuat
9	Lebih penting secara ekstrim
2, 4, 6, 8	Nilai tengah antara angka ganjil
1/x (angka kebalikan)	Nilai jika kepentingannya terbalik

Selain itu, terdapat nilai *Random Consistency Index* berdasarkan banyaknya kriteria yaitu ditunjukkan pada tabel 2. Untuk mencari nilai *Consistency Index (CI)* dan *Consistency Ratio (CR)* maka digunakan persamaan berikut.

$$CI = \lambda \max - n / n - 1 \tag{2}$$

$$CR = CI / RI \tag{3}$$

Criteria	K0	K1	...	Kn-1	Priority Vector
K0	1	X _{0,1}	...	X _{0,n-1}	Sum{(x _{0,i} /t _i):(x _{0,i} /t _{n-1})/n
K1	1/X _{1,0}	1	...	X _{1,n-1}	Sum{(x _{1,i} /t _i):(x _{1,i} /t _{n-1})/n
...	1
Kn-1	1/X _{n-1,0}	1/X _{n-1,1}	...	1	Sum{(x _{n-1,i} /t _i):(x _{n-1,i} /t _{n-1})/n
Total	Sum(x _{0,0} :x _{0,1,0})	Sum(x _{0,1} :x _{0,1,1})	...	Sum(x _{0,n-1} :x _{0,n-1,1})	Sum(PV)

Gbr. 11 Implementasi Matriks AHP

Berdasarkan ilustrasi pada gambar 11, berikut merupakan langkah-langkah implementasi iterasi AHP kedalam program.

- Dibuat matriks dengan ordo N x N dimana N merupakan jumlah kriteria.
- Input perbandingan berpasangan diurutkan ke dalam array lalu mengisi matriks bagian segitiga atas ditunjukkan dengan warna hijau.
- Sedangkan segitiga bawah matriks berisi kebalikan dari tiap elemen segitiga atas.
- Elemen berwarna biru bernilai 1 akibat perbandingan dengan dirinya sendiri.
- Nilai total didapat dari hasil jumlah tiap kolom matriks.
- Nilai PV didapat dari jumlah tiap kolom dikali dengan total dibagi jumlah kriteria.
- Nilai *CI* dan *CR* akan didapat dengan persamaan 2 dan 3.

Selanjutnya pada algoritma SAW memiliki persamaan dalam menormalisasikan nilai menjadi skala yang sama dan menghitung nilai akhir berdasarkan pembobotan yaitu:

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\max x_{ij}} \rightarrow \text{jika } j \text{ adalah benefit} \\ \frac{\min x_{ij}}{x_{ij}} \rightarrow \text{jika } j \text{ adalah cost} \end{cases} \tag{4}$$

$$v_i = \sum_{j=1}^n w_j r_{ij} \tag{5}$$

Dalam implementasi metode SAW, data disusun ke dalam matriks seperti pada gambar 12 dengan langkah berikut.

- Dibuat matrik dengan ordo A x K dimana A merupakan jumlah alternatif dan K merupakan jumlah atribut berdasarkan kriteria.
- Dibuat matrik yang menandakan sifat attribut kriteria yang merupakan cost atau benefit.

Alt/v	K0	K1	...	Kn-1
A0	A ₀ K ₀	A ₀ K ₁	...	A ₀ K _{n-1}
A1	A ₁ K ₀	A ₁ K ₁	...	A ₁ K _{n-1}
...
An-1	A _{n-1} K ₀	A _{n-1} K ₁	...	A _{n-1} K _{n-1}
Type	c/b	c/b	...	c/b

Gbr. 12 Implementasi Matriks SAW

Untuk menormalisasikan tiap elemen, matrik ditranspose untuk menentukan nilai *max/min* berdasarkan *type/sifat* atribut. Kemudian dilakukan operasi seperti gambar 13.

Alt _v	K0	K1	...	Kn-1
A0	$\frac{IF(t_{K_0} = c; \min(K_0)/A_0; K_0; A_0; K_0 / \max(K_0))}{A_0; K_0 / \max(K_0)}$	$\frac{IF(t_{K_1} = c; \min(K_1)/A_0; K_1; A_0; K_1 / \max(K_1))}{A_0; K_1 / \max(K_1)}$...	$\frac{IF(t_{K_{n-1}} = c; \min(K_{n-1})/A_1; K_{n-1}; A_1; K_{n-1} / \max(K_{n-1}))}{A_1; K_{n-1} / \max(K_{n-1})}$
A1	$\frac{IF(t_{K_0} = c; \min(K_0)/A_1; K_0; A_1; K_0 / \max(K_0))}{A_1; K_0 / \max(K_0)}$	$\frac{IF(t_{K_1} = c; \min(K_1)/A_1; K_1; A_1; K_1 / \max(K_1))}{A_1; K_1 / \max(K_1)}$...	$\frac{IF(t_{K_{n-1}} = c; \min(K_{n-1})/A_1; K_{n-1}; A_1; K_{n-1} / \max(K_{n-1}))}{A_1; K_{n-1} / \max(K_{n-1})}$
...
A _{n-1}	$\frac{IF(t_{K_0} = c; \min(K_0)/A_{n-1}; K_0; A_{n-1}; K_0 / \max(K_0))}{A_{n-1}; K_0 / \max(K_0)}$	$\frac{IF(t_{K_1} = c; \min(K_1)/A_{n-1}; K_1; A_{n-1}; K_1 / \max(K_1))}{A_{n-1}; K_1 / \max(K_1)}$...	$\frac{IF(t_{K_{n-1}} = c; \min(K_{n-1})/A_{n-1}; K_{n-1}; A_{n-1}; K_{n-1} / \max(K_{n-1}))}{A_{n-1}; K_{n-1} / \max(K_{n-1})}$

Gbr. 13 Implementasi Normalisasi SAW

Kemudian untuk proses modifikasi dalam pemecahan masalah terkait indeks konsistensi saat melakukan perbandingan berpasangan, maka diterapkan teknik skala terhitung dengan persamaan sebagai berikut:

$$T(\text{total}) = \sum_{i=1}^n K_i \quad (6)$$

$$M(\text{Multiplier}) = \frac{8}{T} \rightarrow T \neq 0 \quad (7)$$

$$\Delta K(\text{Delta Kriteria}) = K_i - K_j \quad (8)$$

$$X_{ij} = \begin{cases} \frac{(\Delta K * M) + 1}{1} \rightarrow \text{jika } \Delta K \geq 0 \\ \frac{1}{(\Delta K * M) + 1} \rightarrow \text{jika } \Delta K < 0 \end{cases} \rightarrow M \neq 0 \quad (9)$$

Pada ilustrasi gambar 9 sebelah kanan merupakan penentuan tingkat kepentingan antar kriteria dengan menggunakan persamaan (6), (7), (8), (9) untuk diubah menjadi skala pada tabel 1.

C. Hasil Pengujian

Pada tahap pengujian, penelitian ini bertujuan untuk mengetahui hasil perbandingan implementasi metode antara program dengan perhitungan manual (*Excel*), hasil perhitungan data ide, kehandalan, serta performa yang didapat.

1) *Pengujian Implementasi AHP*: Tahap pengujian dilakukan dengan iterasi manual menggunakan Excel serta percobaan yang sama menggunakan fungsi program. Dengan persamaan (1) maka didapat 15 masukkan dari 6 kriteria dan masing-masing akan diberi nilai sebagai berikut dengan simbol (K) merupakan singkatan dari kriteria.

- A1 – K1 = 47
- A1 – K2 = 28
- A2 – K1 = 95
- A2 – K2 = 17
- A3 – K1 = 12
- A3 – K2 = 25
- A4 – K1 = 119
- A4 – K2 = 22
- A5 – K1 = 96
- A5 – K2 = 28

Nilai yang dimasukkan mengacu pada skala 1 – 9 metode AHP seperti pada tabel 1.

TABEL III
PENGUJIAN ITERASI AHP PADA PROGRAM

	Excel	Program
PV K1	0,121	0,1213365779
PV K2	0,227	0,2268166929

PV K3	0,329	0,3294533962
PV K4	0,138	0,1377921083
PV K5	0,145	0,1453547561
PV K6	0,039	0,03924646861
CI	0,115	0,1146573026
CR	0,092	0,0924655666

Didapat perbandingan hasil nilai-nilai *Priority Vector*, *Principal Eigen Value* (λ_{max}), *CI* dan *CR* diantara keduanya yang terdapat pada tabel 3. Dari data tersebut dapat dikatakan bahwa tidak ada perbedaan hasil yang menandakan program sudah sesuai dalam mengimplementasikan fungsi iterasi pada metode AHP.

2) *Pengujian Implementasi SAW*: Pengujian program dilakukan dengan menjalankan normalisasi pada atribut data kumpulan alternatif menggunakan persamaan 4. Pengujian menggunakan 5 alternatif dengan 2 atribut sebagai *cost* dan *benefit*.

- A1 – K1 = 47
- A1 – K2 = 28
- A2 – K1 = 95
- A2 – K2 = 17
- A3 – K1 = 12
- A3 – K2 = 25
- A4 – K1 = 119
- A4 – K2 = 22
- A5 – K1 = 96
- A5 – K2 = 28

Hasil pengujian normalisasi program dibandingkan dengan perhitungan manual menggunakan *Excel* dapat dilihat pada tabel 4. Keduanya menghasilkan nilai yang sama menggunakan persamaan (4). Maka dapat dikatakan bahwa program sudah mengimplementasikan algoritma normalisasi dengan metode SAW dengan benar.

TABEL IV
PENGUJIAN NORMALISASI SAW PADA PROGRAM

	Excel	Program
A1 K1	0,26	0,2553191489
A1 K2	1,00	1
A2 K1	0,13	0,1263157895
A2 K2	0,61	0,6071428571
A3 K1	1,00	1
A3 K2	0,89	0,8928571429
A4 K1	0,10	0,1008403361
A4 K2	0,79	0,7857142857
A5 K1	0,13	0,125
A5 K2	1,00	1

3) *Pengujian Kehandalan AHP Skala Terhitung*: Tahap pengujian terhadap kehandalan antara Teknik standart dengan Teknik skala terhitung adalah membandingkan peluang terjadinya ketidakkonsistenan *input* ketika menentukan tingkat kepentingan antar kriteria. Prosedur yang dilakukan adalah menguji persentase terjadinya *input* konsisten menggunakan 3 kriteria. Akan diambil kombinasi nilai unik untuk dijadikan *input* dan menghitung jumlah hasil $CR \leq 0.1$ dari seluruh percobaan berdasarkan persamaan (2) dan (3) setelah dilakukan iterasi.

Dari data tabel 5 membuktikan bahwa untuk jumlah 3 kriteria hanya menghasilkan 17,484% *input* yang dinyatakan

TABEL VI
DATA ATRIBUT IDEA

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16
A1	113	19	7	0,23	1	19	7	1	8	4	6	3	2	1	8	1
A2	56	13	4	0,30	1	13	4	1	2	2	1	1	1	1	2	1
A3	47	28	12	0,85	1	28	12	1	6	2	5	1	3	1	12	1
A4	159	10	7	0,11	2	23	13	1	2	2	1	1	1	1	4	1
A5	90	13	6	0,21	2	23	13	1	4	2	4	1	4	2	6	2
A6	95	17	8	0,26	3	55	26	2	3	3	1	3	2	3	8	1
A7	12	25	13	3,17	3	55	26	2	6	1	6	2	1	1	10	1
A8	112	13	5	0,16	3	55	26	2	6	1	6	2	2	1	7	1
A9	82	14	5	0,23	4	75	40	3	5	2	3	1	1	1	7	1
A10	119	22	10	0,27	4	75	40	3	4	2	4	1	2	1	9	2
A11	96	28	18	0,48	4	75	40	3	6	3	1	1	3	1	18	2
A12	13	11	7	1,38	4	75	40	3	8	2	8	3	2	3	7	2
A15	70	20	4	0,34	1	20	4	1	8	2	8	2	3	2	6	1
A14	176	11	5	0,09	1	11	5	1	4	1	4	2	2	1	5	2
A15	60	15	4	0,32	1	15	4	1	3	3	3	2	3	2	9	1

konsisten. Sedangkan dengan teknik skala terhitung dapat menghasilkan tingkat konsistensi *input* sebesar 99,98% pada kasus yang sama.

TABEL V
HASIL PELUANG INPUT KONSISTEN

	Berpasangan	Skala Terhitung
Jumlah Kriteria	3	3
Skala	17	17
Jumlah Percobaan	4913	4913
Konsisten	859	4913
Tidak Konsisten	4054	1
Persentase	17,484%	99,98%

4) *Pengujian Sistem Pendukung Keputusan*: Pada tahap ini akan dilakukan pengujian hasil guna menentukan ide terbaik berdasarkan nilai *input* preferensi pada kriteria yang memiliki nilai berbasis data [9]. Data *input* yang digunakan untuk melakukan pengujian terdiri dari kategori utama dan sub-kategori dengan skala 0-100 yang menjadi preferensi seorang *innovation manager*. Berikut adalah nilai dari setiap data *input* yang digunakan.

- K1. Keterbaruan = 5
- K2. *Trending* = 80
 - Sub k1. *Like* = 90
 - Sub k2. *Comment* = 40
 - Sub k3. *Bounce Rate* = 0
- K3. Profil Kreator = 70
 - Sub k1. Total *Idea* = 90
 - Sub k2. Total *Like* = 40
 - Sub k3. Total *Comment* = 0
 - Sub k4. Total *Achievement* = 0
- K4. *Lean Canvas* = 10
 - Sub k1. *Customer* = 10
 - Sub k2. *Problem* = 80

- Sub k3. *Early Adopter* = 10
- Sub k4. *Existing Solution* = 10
- Sub k5. *Unique Value* = 10
- Sub k6. *Proposed Solution* = 10
- K5. Jumlah Tim = 70
- K6. Jumlah *Attachment* = 20

Tabel 6 merupakan 15 data ide beserta nilai atributnya yang sudah ada pada sistem IdeaBox. Berikut adalah keterangan untuk setiap simbolnya.

Simbol A = Alternatif
A1 = Friendly Parking
A2 = E-Wallet Payment
A3 = Daily Standup
A4 = Mobility Car
A5 = Office Computer
A6 = Eventeer
A7 = Retro Event
A8 = Annual Donation
A9 = Weekly Work Check
A10 = People Backup
A11 = PeDePe
A12 = Annual Bazaar
A13 = Free Launch
A14 = Achievement to Reward
A15 = Coordination Ceremony

Simbol X = Atribut
X1 = Keterbaruan
X2 = Like
X3 = Comment
X4 = Bounce Rate
X5 = Ide Kreator
X6 = Like Kreator
X7 = Comment Kreator
X8 = Achievement
X9 = Customer
X10 = Problem
X11 = Early Adopter
X12 = Existing Solution
X13 = Unique Value
X14 = Proposed Solution
X15 = Jumlah Tim
X16 = Jumlah Lampiran

Mencari *Priority Vector*, *CI* dan *CR* untuk tingkatan *main* kriteria menggunakan iterasi AHP seperti pada tabel 7.

TABEL VII
ITERASI AHP MAIN KRITERIA

Main	K1	K2	K3	K4	K5	K6	PV
K1	1,0	0,3	0,3	0,9	0,3	0,7	0,1
K2	3,4	1,0	1,3	3,2	1,3	2,9	0,3
K3	3,0	0,8	1,0	2,9	1,0	2,6	0,2

K4	1,2	0,3	0,3	1,0	0,3	0,8	0,1	
K5	3,0	0,8	1,0	2,9	1,0	2,6	0,2	
K6	1,5	0,3	0,4	1,3	0,4	1,0	0,1	
Total	13,1	3,5	4,4	12,1	4,4	10,5	1,0	
PEV								6,015
CI								0,003
CR								0,002

Pada Tabel 7 menunjukkan bahwa nilai CR 0,2% tidak lebih dari 10%. Maka bisa dikatakan *input* sudah konsisten. Selanjutnya adalah melakukan iterasi pada sub kriteria dari masing-masing *main* kriteria yang memiliki turunan. Nilai PV dari *main* kriteria akan menjadi acuan untuk nilai PV pada iterasi sub kriteria.

TABEL VIII
ITERASI AHP SUB KRITERIA TRENDING

Sub Trending	k1	k2	k3	PV	
k1	1,0	4,1	6,5	0,7	
k2	0,2	1,0	3,5	0,2	
k3	0,2	0,3	1,0	0,1	
Total	1,4	5,4	11,0	1,000	
PEV					3,106
CI					0,053
CR					0,092

Pada Tabel 8 menunjukkan bahwa nilai CR 9,2% tidak lebih dari 10%. Maka bisa dikatakan *input* sudah konsisten.

TABEL IX
ITERASI AHP SUB KRITERIA PROFIL KREATOR

Sub Kreator	k1	k2	k3	k4	PV
k1	1,0	4,1	4,1	4,1	0,6
k2	0,2	1,0	1,0	1,0	0,1
k3	0,2	1,0	1,0	1,0	0,1

k4	0,2	1,0	1,0	1,0	0,1	
Total	1,7	7,1	7,1	7,1	1,0	
PEV						4,0
CI						0,0
CR						0,0

Pada Tabel 9 menunjukkan bahwa nilai CR 0% tidak lebih dari 10%. Maka bisa dikatakan *input* sudah konsisten.

TABEL X
ITERASI AHP SUB KRITERIA LEAN CANVAS

Sub LC	k1	k2	k3	k4	k5	k6	PV	
k1	1,0	0,2	1,0	1,0	1,0	1,0	0,1	
k2	5,3	1,0	5,3	5,3	5,3	5,3	0,5	
k3	1,0	0,2	1,0	1,0	1,0	1,0	0,1	
k4	1,0	0,2	1,0	1,0	1,0	1,0	0,1	
k5	1,0	0,2	1,0	1,0	1,0	1,0	0,1	
k6	1,0	0,2	1,0	1,0	1,0	1,0	0,1	
Total	10,3	1,9	10,3	10,3	10,3	10,3	1,0	
PEV								6,00
CI								0,00
CR								0,00

Pada Tabel 10 menunjukkan nilai CR 0% tidak lebih dari 10%. Maka bisa dikatakan *input* sudah konsisten. Lalu nilai tiap PV pada sub kriteria dikalikan dengan PV kriteria utamanya. Pada tabel 11 merupakan akumulasi perkalian nilai PV antara *main* kriteria dengan sub kriteria menjadi nilai pembobotan (W).

TABEL XI
PERHITUNGAN PEMBOBOTAN

K1	K2			K3			
	k1	k2	k2	k1	k2	k2	k4
0,075	0,193	0,063	0,024	0,134	0,033	0,033	0,033
W1	W2	W3	W4	W5	W6	W7	W8

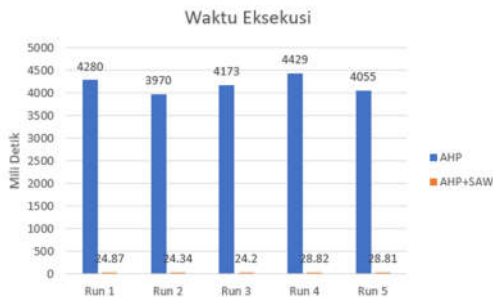
TABEL XII
NORMALISASI SAW PADA ATRIBUT ALTERNATIF

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	Skor
A1	0,11	0,68	0,39	0,07	0,25	0,25	0,18	0,33	1,00	1,00	0,75	1,00	0,50	0,33	0,44	0,50	0,447
A2	0,21	0,46	0,22	0,10	0,25	0,17	0,10	0,33	0,25	0,50	0,13	0,33	0,25	0,33	0,11	0,50	0,282
A3	0,26	1,00	0,67	0,27	0,25	0,37	0,30	0,33	0,75	0,50	0,63	0,33	0,75	0,33	0,67	0,50	0,575
A4	0,08	0,36	0,39	0,03	0,50	0,31	0,33	0,33	0,25	0,50	0,13	0,33	0,25	0,33	0,22	0,50	0,331
A5	0,13	0,46	0,33	0,07	0,50	0,31	0,33	0,33	0,50	0,50	0,50	0,33	1,00	0,67	0,33	1,00	0,442
A6	0,13	0,61	0,44	0,08	0,75	0,73	0,65	0,67	0,38	0,75	0,13	1,00	0,50	1,00	0,44	0,50	0,532
A7	1,00	0,89	0,72	1,00	0,75	0,73	0,65	0,67	0,75	0,25	0,75	0,67	0,25	0,33	0,56	0,50	0,695
A8	0,11	0,46	0,28	0,05	0,75	0,73	0,65	0,67	0,75	0,25	0,75	0,67	0,50	0,33	0,39	0,50	0,458
A9	0,15	0,50	0,28	0,07	1,00	1,00	1,00	1,00	0,63	0,50	0,38	0,33	0,25	0,33	0,39	0,50	0,535
A10	0,10	0,79	0,56	0,08	1,00	1,00	1,00	1,00	0,50	0,50	0,50	0,33	0,50	0,33	0,50	1,00	0,682
A11	0,13	1,00	1,00	0,15	1,00	1,00	1,00	1,00	0,75	0,75	0,13	0,33	0,75	0,33	1,00	1,00	0,882
A12	0,92	0,39	0,39	0,44	1,00	1,00	1,00	1,00	1,00	0,50	1,00	1,00	0,50	1,00	0,39	1,00	0,658
A15	0,17	0,71	0,22	0,11	0,25	0,27	0,10	0,33	1,00	0,50	1,00	0,67	0,75	0,67	0,33	0,50	0,404
A14	0,07	0,39	0,28	0,03	0,25	0,15	0,13	0,33	0,50	0,25	0,50	0,67	0,50	0,33	0,28	1,00	0,346
A15	0,20	0,54	0,22	0,10	0,25	0,20	0,10	0,33	0,38	0,75	0,38	0,67	0,75	0,67	0,50	0,50	0,409

K4						K5	K6
k1	k2	k3	k4	k5	k6		
0,008	0,042	0,008	0,008	0,008	0,008	0,232	0,099
W9	W10	W11	W12	W13	W14	W15	W16

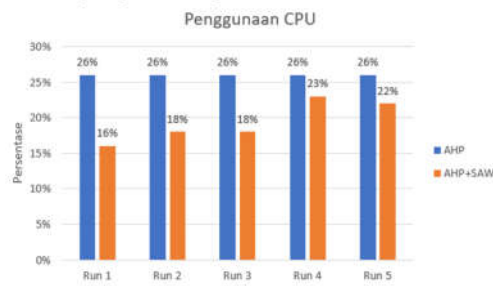
Untuk mendapatkan nilai skor pada tabel 12 maka nilai setiap atribut alternatif diakumulasikan dengan nilai W pada tabel 11 menggunakan persamaan (5). Angka skor ini akan menentukan ranking untuk setiap alternatif yang ada berdasarkan skor tertinggi. Hasil yang diperoleh merupakan alternatif **A11 (PeDePe)** dengan skor 0,882 dan urutan kedua adalah **A7 (Retro Event)** dengan skor 0,695. Hasil yang diperoleh sama dengan hasil yang didapatkan menggunakan aplikasi seperti pada gambar 10.

5) *Pengujian Performa AHP dan kombinasi AHP SAW:* Pada tahap ini akan dilakukan pengujian performa antara penggunaan AHP keseluruhan maupun penggunaan kombinasi AHP dan SAW dalam menangani jumlah data yang banyak. Percobaan dilakukan bertahap sebanyak 5 kali menggunakan 500 data dan mempertimbangkan aspek waktu eksekusi, penggunaan *processor* serta penggunaan *memory*. Jumlah data tersebut merupakan hasil duplikasi dari data yang sudah ada. Meskipun data yang digunakan merupakan data duplikasi, namun data tersebut tetap dapat digunakan dalam uji performa. Tidak ada perbedaan karakteristik dengan data yang bersifat unik dikarenakan semua data memiliki jumlah atribut yang sama.



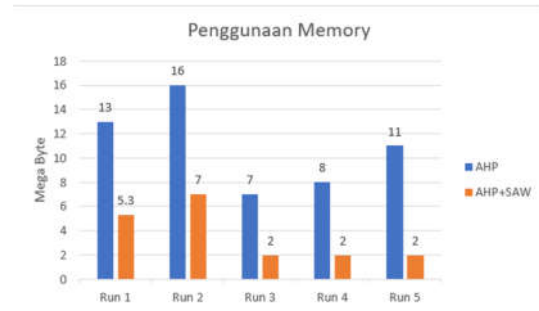
Gbr. 14 Performa Berdasarkan Waktu Eksekusi

Pada gambar 14 terlihat perbedaan waktu eksekusi yang sangat signifikan. Kombinasi metode SAW dapat mempersingkat proses perhitungan karena tidak memerlukan proses iterasi yang berulang.



Gbr. 15 Performa Berdasarkan Pemakaian CPU

Dari segi penggunaan *processor* pada gambar 15 ada perbedaan dimana kombinasi AHP dan SAW memerlukan sumber pemrosesan yang lebih sedikit namun tidak terlalu signifikan.



Gbr. 16 Performa Berdasarkan Pemakaian Memori

Dalam penggunaan *memory* seperti pada gambar 16, keduanya relatif tidak membutuhkan kapasitas yang besar karena dari data yang tertera penggunaan *memory* tertinggi baru mencapai angka 16 MB.

IV. KESIMPULAN

Hasil dari penelitian ini berfokus pada pengujian dari penerapan metode AHP dan SAW dalam mendukung suatu keputusan pada pemilihan ide terbaik berdasarkan kriteria. Pengujian yang dilakukan meliputi uji implementasi program dengan menyocokkan hasil pada perhitungan manual, uji tingkat konsistensi *input* pada AHP, hingga uji performa eksekusi iterasi ketika program dijalankan. Berikut ini adalah kesimpulan yang diperoleh dari hasil penelitian ini yaitu:

- 1) Proses iterasi data pada algoritma *Analytical Hierarchy Process* dan normalisasi algoritma *Simple Additive Weighting* dapat diterapkan pada program dengan hasil yang sesuai pada perhitungan manual.
- 2) Penerapan metode AHP dan SAW dalam mendukung pemilihan ide terbaik dapat dilakukan pada kriteria berbasis data kuantitatif.
- 3) Modifikasi metode AHP dalam menentukan nilai perbandingan berpasangan dapat dilakukan karena menghasilkan keluaran yang sama namun dengan peluang konsistensi yang lebih besar. Untuk kasus perbandingan 3 kriteria sistem AHP biasa hanya berpeluang konsisten sebanyak 17,484%. Dengan skala terhitung, peluang konsisten menjadi 99,98% dalam kasus yang sama.
- 4) Dengan modifikasi metode AHP membuat kombinasi tingkat kepentingan kriteria menjadi terbatas. Namun hal tersebut yang menghindari terjadinya kasus *input* yang tidak konsisten.
- 5) Penerapan kombinasi metode AHP dan SAW cenderung lebih baik dibandingkan hanya menggunakan metode AHP karena menggunakan sumber daya yang lebih sedikit dan tidak terlalu

berpengaruh terhadap waktu ketika dihadapkan dengan data yang banyak.

V. SARAN

Aplikasi dalam penelitian ini dapat dikembangkan lebih lanjut sehingga tidak hanya memperhitungkan data-data matematis saja namun juga dapat mengklasifikasikan *feedback* pengguna pada analisis tingkat sentimen. Penelitian ini mengimplementasikan metode AHP dan SAW masih pada sisi *front-end* dimana pengembangan sistem akan lebih ideal jika dilakukan pada sisi *back-end*. Kekurangan fitur dan kesalahan program juga tidak luput untuk terus dapat diperbaiki seperti manajemen sumber daya yang lebih baik hingga penerapan *best practice* yang lebih ditingkatkan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terimakasih yang sebesar-besarnya kepada:

- 1) Allah SWT atas segala nikmat, rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan penelitian ini. ng tua dan Putri Syalsabila selaku saudara sekeluarga.
- 2) Bapak Mujiono, Ibu Hidayatul Mufidah selaku orang tua dan Putri Syalsabila selaku saudara keluarga yang selalu memberikan do'a.
- 3) Indry Puji Lestari atas semangat dan do'anya yang memberikan motivasi kuat untuk terus memiliki keyakinan.
- 4) Bapak. Aditya Prapanca S.T., M.Kom. selaku dosen pembimbing yang telah membantu dan memudahkan proses penyelesaian penelitian.
- 5) Seluruh teman-teman Jurusan Teknik Informatika yang memberi semangat dan motivasi.

DAFTAR PUSTAKA

- [1] Ashaf, D. H., Hidayat, S. W., & Ahmadi. (2019). Decision Support System Determines The Purchase Of House Right Using Analytical Hierarchy Process (Ahp) And Borda Methods, 10(1). *International Journal Of Asro*, 1-9.
- [2] Banken, V., Ilmer, Q., Seeber, I., & Haeussler, S. (2019). A Method For Smart Idea Allocation In Crowd-Based Idea Selection, 124. *Decision Support Systems*, 1-10.

- [3] Daniawan, B. (2018). Evaluation Of Lecturer Teaching Performance Using Ahp And Saw Methods, 1(2). *Bit-Tech*, 30-39.
- [4] Díaz, R. P., Arroyo, J. C., & Barrera, F. V. (2021). Metacognitive Feelings As A Source Of Information In The Evaluation And Selection Of Creative Ideas, 39. *Thinking Skills And Creativity*, 1-9.
- [5] Ermin, Sunardi, & Fadil, A. (2019). Metode Simple Additive Weighting Pada Penentuan Penerimaan Karyawan, 8(2). 125-131.
- [6] Gerlach, S., & Brem, A. (2017). Idea Management Revisited: A Review Of The Literature And Guide For Implementation, 1. *International Journal Of Innovation Studies*, 144-161.
- [7] Gill, O. (2018). *Using React Native For Mobile Software Development*. Metropolis University.
- [8] Mikelsone, E., Pilbergs, A., & Segers, J. -P. (2021). Benefits Of Web-Based Idea Management System Application, 29(3). *European Journal Of Management Issues*, 151-161.
- [9] Mikelsone, E., Volkova, T., & Lielā, E. (2019). Practical Evidence Of Web-Based Idea Management Systems: Classification And Application, 2. *Research For Rural Development*, 276-283.
- [10] Munthafa, A. E., & Mubarak, H. (2017). Penerapan Metode Analytical Hierarchy Process Dalam Sistem Pendukung Keputusan Penentuan Mahasiswa Berprestasi, 3(2). *Jurnal Siliwangi*, 192-201.
- [11] Nursaid, F. F., Brata, A. H., & Kharisma, A. P. (2020). Pengembangan Sistem Informasi Pengelolaan Persediaan Barang Dengan Reactjs Dan React Native Menggunakan Prototype (Studi Kasus : Toko Uda Fajri), 4(1). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 46-55.
- [12] Sahir, S. H., Rosmawati, R., & Minan, K. (2017). Simple Additive Weighting Method To Determining Employee Salary Increase Rate, 3(8). *International Journal Of Scientific Research In Science And Technology*, 42-48.
- [13] Supriadi, A., Rustandi, A., Komarlina, D. H., & Ardiani, G. T. (2018). *Analytical Hierarchy Process (Ahp) Teknik Penentuan Strategi Daya Saing Kerajinan Bordir*. Sleman: Deepublish.
- [14] Syam, S., & Rabidin, M. (2019). Metode Simple Additive Weighting Dalam Sistem Pendukung Keputusan Pemilihan Karyawan Berprestasi (Studi Kasus : Pt. Indomarco Prismatama Cabang Tangerang 1), 6(1). *Jurnal Keilmuan Dan Aplikasi Teknik*, 14-18.
- [15] Kurniawan, H., Apriliah, W., Kurniawan, I., & Firmansyah, D. (2020). Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Penggajian Pada Smk Bina Karya Karawang, 14(4). *Interkom*, 159-169