

Sistem Pendukung Keputusan Penentuan Dosen Pembimbing Tugas Akhir Menggunakan Fuzzy dan *Simple Additive Weighting* Berbasis Android: Studi Kasus IKADO Surabaya

T Rahmawati¹, A Wirapraja², E C Soesilo³

^{1,2,3}Institut Informatika Indonesia

E-mail: tita@ikado.ac.id¹, alex@ikado.ac.id², tjiauesterz@gmail.com³

Abstrak. Tugas Akhir adalah suatu karya ilmiah yang dilakukan dengan cara melakukan penelitian berdasarkan suatu masalah dan didukung data yang diperoleh dari kegiatan penelitian. Dalam proses menyelesaikan tugas akhir, biasanya mahasiswa dibimbing oleh dosen pembimbing. Saat ini mahasiswa di IKADO Surabaya dalam memilih dosen pembimbing hanya dengan penilaian secara subjektif dan hanya melalui intuisi saja. Hal ini menyebabkan dosen pembimbing yang dipilih oleh mahasiswa kurang relevan dengan topik tugas akhir dan juga dari segi pengalaman membimbing dengan topik tugas akhir tersebut masih sedikit sehingga mempengaruhi hasil / kualitas dari tugas akhir itu sendiri. Oleh sebab itu dibutuhkan suatu aplikasi yang dapat memberikan rekomendasi kepada mahasiswa terkait dosen pembimbing tugas akhirnya sesuai dengan kriteria-kriteria tertentu. Dalam penelitian ini, aplikasi yang dikembangkan akan menggunakan metode *fuzzy* dan *simple additive weighting* untuk mendapatkan rekomendasi dosen pembimbing sesuai topik tugas akhir mahasiswa. Hasil tersebut menampilkan pilihan nama dosen pembimbing pertama dan kedua, beserta lima peringkat teratas nama dosen yang dapat dipakai sebagai alternatif pembimbing. Metode pengembangan yang digunakan yaitu *extreme programming*, serta menggunakan pengujian berupa *blackbox testing* sedangkan untuk uji coba kemudahan dan kegunaan aplikasi menggunakan pengujian *usability testing*. Dengan dibuatnya aplikasi pendukung keputusan yang dinamakan SIPEKA ini, dapat membantu mahasiswa dan koordinator tugas akhir dalam menentukan alternatif dosen pembimbing tugas akhir, dengan memanfaatkan fitur-fitur yang terdapat pada aplikasi.

Kata kunci: Sistem Pendukung Keputusan; *Fuzzy*; *Simple Additive Weighting*; Android.

Abstract. *Final Project is a scientific work carried out by conducting research based on a problem and supported by data obtained from research activities. In the process of completing the final project, students are usually guided by a supervisor. Currently, students at IKADO Surabaya choose their supervisor only by subjective assessment and only through intuition. This causes the supervisors chosen by students to be less relevant to the topic of the final project and also in terms of experience in guiding with the topic of the final project is still small so that it affects the results / quality of the final project itself. Therefore, we need an application that can provide recommendations to students regarding their final assignment supervisor in accordance with certain criteria. In this study, the application developed will use the fuzzy method and simple additive weighting to get the supervisor's recommendation according to the topic of the student's final project. The results show the choice of the names of the first and second supervisors, along with the top five ranking lecturers' names that can be used as alternative supervisors. The development*

method used is extreme programming, and uses testing in the form of Blackbox Testing, while testing the ease and usability of the application uses usability testing. By making a decision support application called SIPEKA, it can help students and final project coordinators in determining alternative final project supervisors, by utilizing the features contained in the application.

Keywords: final project; Fuzzy; Simple Additive Weighting; android.

1. Pendahuluan

Tugas akhir adalah suatu karya yang ditulis oleh mahasiswa dari hasil penelitian yang mengangkat masalah sesuai dengan bidang ilmu dari program studi mahasiswa tersebut. Tugas akhir merupakan kegiatan yang wajib diambil oleh mahasiswa sebagai syarat kelulusan. Untuk membantu menentukan judul, topik penelitian, arah penelitian dan juga dalam format penulisan tugas akhir maka mahasiswa akan diarahkan oleh satu atau dua dosen pembimbing [1]. Seorang dosen pembimbing memiliki tugas dan peran yang penting dalam penyelesaian tugas akhir mahasiswa [2]. Di IKADO Surabaya, pihak yang menentukan dosen pembimbing adalah mahasiswa yang bersangkutan. Mahasiswa akan memilih dosen pembimbing yang menurutnya cocok atau sesuai dengan topik penelitian yang diambil. Namun hal tersebut akan menjadi tidak efektif jika mahasiswa tidak mendapatkan informasi yang cukup terkait kapabilitas dan pengalaman dari dosen pembimbing yang dipilih. Dalam penelitian oleh Amandus dkk [3] melakukan penelitian dengan sampel sebanyak 82 mahasiswa Poltekkes Kemenkes Pontianak hasilnya menunjukkan bahwa adanya ketidakpuasan mahasiswa dalam pemberian pendampingan kepada mahasiswa dalam penyelesaian tugas akhir. Jika mahasiswa memilih dosen pembimbing kurang tepat dengan topik penelitiannya dan juga kurang berpengalaman dalam bidang yang akan diteliti oleh mahasiswa. Hal tersebut mengakibatkan hasil dan kualitas tugas akhir akan menjadi tidak baik.

Di Institut Informatika Indonesia Surabaya, penunjukan dosen pembimbing tugas akhir di lingkungan Fakultas Teknologi Informasi masih dilakukan hanya sebatas pengetahuan dan penilaian mahasiswa yang bersangkutan saja artinya mahasiswa menunjuk dosen pembimbing sesuai dengan bidang kompetensi dosen yang sudah diinformasikan dekan Fakultas sebelumnya. Oleh sebab itu dibutuhkan sistem pendukung keputusan yang berguna untuk menentukan dosen pembimbing tugas akhir yang ditujukan bagi mahasiswa IKADO Surabaya.

Sistem Pendukung Keputusan merupakan hasil dari proses *intelligence*, *design*, dan *choice* [5]. *Intelligence* adalah proses / fase yang dihasilkan dari sistem dengan menggunakan metode sistem cerdas. Sementara *design* adalah fase hasil dari *intelligence* berupa tindakan yang dijadikan menjadi sebuah alternatif. Lalu *choice* adalah fase untuk melakukan sebuah pemilihan yang tepat berdasarkan alternatif yang ada pada fase *design*.

Sehingga dapat disimpulkan bahwa suatu Sistem Pendukung Keputusan adalah suatu metode untuk menyelesaikan suatu masalah dengan mengumpulkan sejumlah fakta yang sesuai dan relevan dari alternatif yang dihadapi kemudian dilakukan pengambilan keputusan atau tindakan secara tepat. Sistem Pendukung keputusan dapat membantu organisasi khususnya pada lini manajerial untuk mendukung manajer mengambil keputusan. Dalam implementasinya Sistem Pendukung Keputusan menggunakan bantuan sistem lain seperti kecerdasan buatan, sistem pakar, atau *fuzzy logic*. Hasil akhir dari sistem pendukung keputusan diperoleh dari proses perangkungan yaitu penjumlahan dari perkalian matriks ternormalisasi R dengan vector bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik (A_i) sebagai solusi [6].

Dengan adanya sistem ini, dapat memberikan panduan agar mahasiswa mendapatkan rekomendasi siapa saja dosen pembimbing yang sesuai topik akan diteliti dengan lebih akurat. Sistem ini dibuat berdasarkan informasi yang diperoleh dari pakar / dosen di IKADO dimana informasi tersebut berupa bidang kompetensi, lama mengajar (berapa tahun mengajar), dan jumlah bimbingan tugas akhir. Informasi-informasi tersebut dalam bahasan selanjutnya juga disebut sebagai kriteria.

Sistem ini menggunakan metode *fuzzy logic* untuk menghitung derajat keanggotaan setiap penilaian kompetensi dosen dan metode *Simple Additive Weighting* yang dapat mencari bobot nilai terbesar dari

semua atribut yang diusulkan, kemudian melakukan perankingan untuk menghasilkan alternatif yang optimal. Dan dari perhitungan tersebut akan menghasilkan informasi berupa penentuan dosen pembimbing yang akurat dan tepat sesuai dengan kriteria-kriteria yang ditentukan sebelumnya oleh peneliti dan berdasarkan informasi yang didapatkan dari Pakar. Sistem serupa juga dirancang oleh Laengge [3], Sistem Pendukung Keputusan dengan metode *Attribute Decision Making* (FMADM) dan *Simple Additive Weighting* (SAW) dengan menggunakan kriteria pendidikan, fungsional, kompetensi, dan kuota atau jumlah bimbingan. Dari penelitian Laengge didapatkan rekomendasi terbaik terkait penentuan dosen pembimbing. Selain Laengge juga dilakukan penelitian yang sejenis yaitu Sistem Pendukung Keputusan dengan metode SAW, dalam penelitian Irawati [4] menggunakan menggunakan 5 variabel kriteria pada pemilihan dosen pembimbing. Lima kriteria tersebut yaitu kriteria pendidikan, golongan, bidang, pengalaman pembimbing, dan beban bimbingan. Dari kriteria tersebut akan ditentukan tiga dosen terbaik untuk menjadi dosen 1, dosen pembimbing 2, dan pembahas.

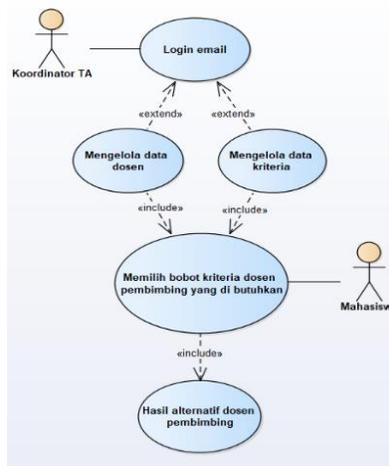
Sistem pendukung keputusan pada penelitian ini menggunakan *mobile based* yaitu android membuat sistem ini menjadi lebih cepat dan lebih praktis untuk digunakan pengguna atau mahasiswa. Metode pengembangan yang digunakan yaitu metode *extreme programming*, dengan tujuan berfokus pada perkembangan yang lebih cepat, serta merespon lebih cepat dengan perubahan yang diminta oleh *user*.

2. Metode

Sebelum dilakukan implementasi program perlu dilakukan analisa dan desain sistem untuk mempermudah implementasi program.

2.1. Use Case Diagram

Use Case Diagram adalah suatu diagram yang digunakan untuk menggambarkan interaksi *user* dengan sistem. *Use case diagram* juga digunakan untuk membantu pengembang aplikasi untuk menentukan kebutuhan fungsional dari sistem [14]. *Use case diagram* pada aplikasi SIPEKA dapat dilihat pada Gambar 1.

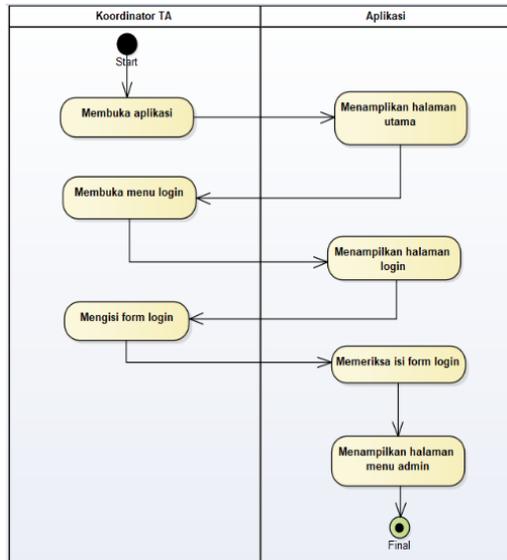


Gambar 1. *Use Case Diagram* Aplikasi SIPEKA

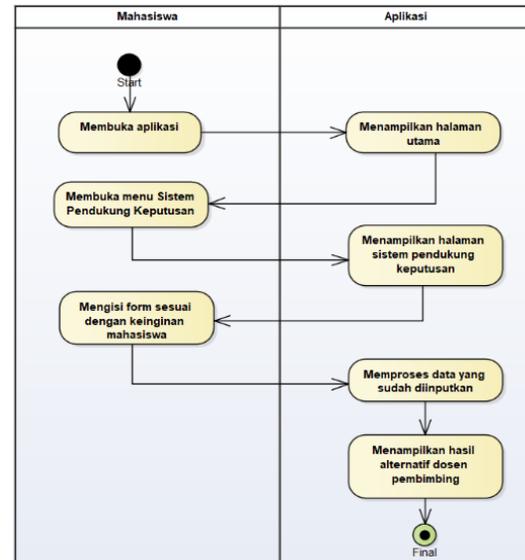
Pada Gambar 1 dapat terlihat bahwa terdapat dua aktor yaitu koordinator dan mahasiswa. Koordinator dapat melakukan *login* terlebih dahulu, setelah itu dapat mengelola data dosen atau data kriteria. Sedangkan mahasiswa dapat menggunakan aplikasi tersebut untuk menentukan bobot kriteria dari setiap kriteria yang ada.

2.2. Activity Diagram

Activity diagram adalah salah satu pemodelan sistem yang merepresentasi grafis dari kegiatan. Fungsi dari diagram ini adalah menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses [14]. Activity diagram login email dan memilih bobot kriteria dosen yang dibutuhkan dapat dilihat pada Gambar 2 dan 3.



Gambar 2. Activity Diagram Login Email



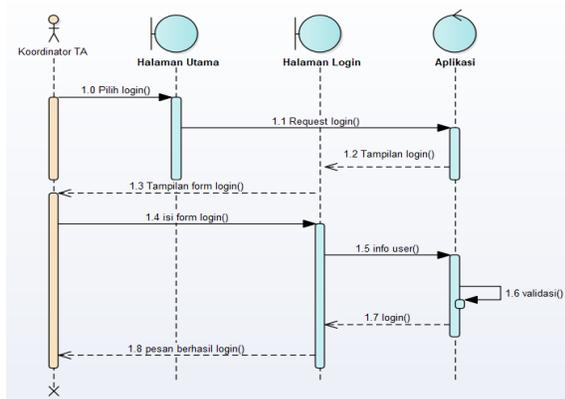
Gambar 3. Activity Diagram Memilih Bobot Kriteria Dosen Yang Dibutuhkan

Pada gambar 2 *Activity Diagram* memilih bobot kriteria dosen yang dibutuhkan dapat dilihat ada dua buah *swimlane* yaitu mahasiswa dan aplikasi. Aktivitas yang terdapat didalamnya yaitu dimulai dari setelah mahasiswa membuka aplikasi, kemudian aplikasi menampilkan halaman utama, lalu mahasiswa memilih menu sistem pendukung keputusan untuk menentukan dosen pembimbing tugas akhir. Kemudian aplikasi menampilkan halaman sistem pendukung keputusan. Setelah itu, mahasiswa mengisi *form* kriteria penentuan dosen pembimbing sesuai yang diinginkan, lalu aplikasi memproses data tersebut dan setelah itu menampilkan hasil alternatif dosen pembimbing.

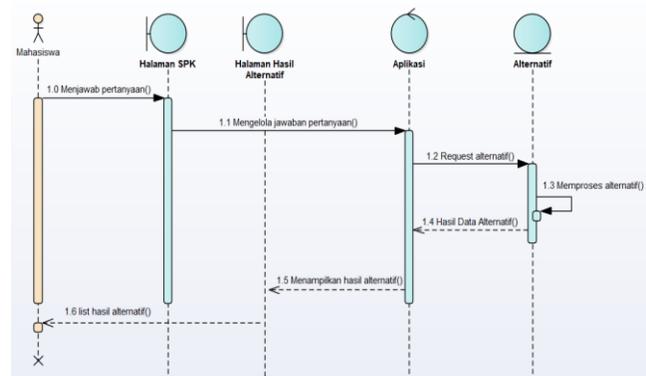
2.3. Sequence Diagram

Sequence Diagram adalah diagram yang menggambarkan secara detail rangkaian pesan yang dikirimkan antar objek dalam sebuah sistem. Diagram ini berfokus pada waktu dan urutan interaksi secara visual dengan menggunakan sumbu vertikal untuk mewakili waktu dan sumbu *horizontal* untuk mewakili objek-objek yang terkait [14].

Pada Gambar 4 *Sequence Diagram login email* terdapat aktor yaitu koordinator yang memilih *login* pada halaman utama, kemudian proses *request login* tersebut di teruskan aplikasi, sehingga dapat menampilkan halaman *login*, kemudian koordinator mengisi *form login* tersebut untuk di kirimkan ke aplikasi melalui halaman *login*, lalu aplikasi melakukan validasi untuk mengecek data *login* dan menampilkan pesan berhasil *login*.



Gambar 4. Sequence Diagram Login Email

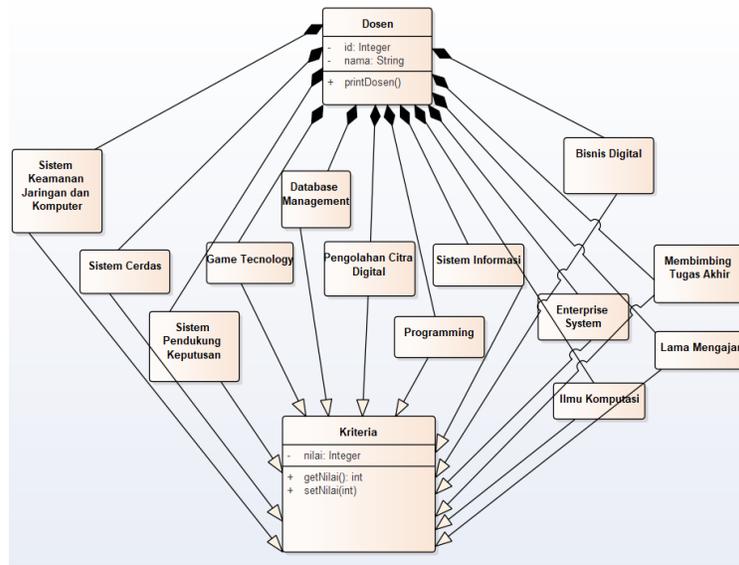


Gambar 5. Sequence Diagram Memilih Bobot Kriteria Dosen

Pada Gambar 5 Sequence Diagram memilih bobot kriteria terdapat aktor yaitu mahasiswa yang menjawab pertanyaan yang telah disediakan pada halaman sistem pendukung keputusan tersebut. Kemudian mengelola jawaban tersebut pada aplikasi, kemudian aplikasi me-request tabel alternatif dan aplikasi memproses tabel alternatif. Setelah itu menampilkan hasil alternatif dan list hasil alternatif.

2.4. Class Diagram

Class diagram dari aplikasi ini digunakan untuk menjelaskan lebih jelas tentang hubungan antara obyek-obyek class pada aplikasi ini [14]. Berikut adalah class diagram yang telah dibuat untuk menjelaskan class yang terdapat pada aplikasi ini.

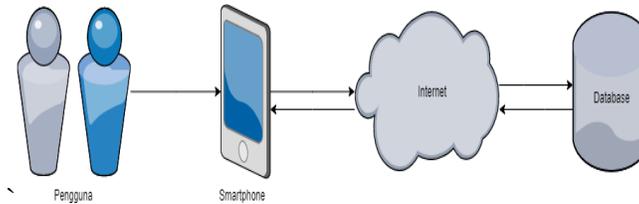


Gambar 6. Class Diagram Aplikasi SIPEKA

Pada Gambar 6 dapat dilihat terdapat 14 buah class pada class diagram tersebut. Class tersebut adalah Dosen, Kriteria, Sistem Keamanan Jaringan dan Komputer, Sistem Cerdas, Sistem Pendukung Keputusan, Game Technology, Database Management, Pengolahan Citra Digital, Programming, Sistem Informasi, Bisnis Digital, Rekayasa Perangkat Lunak, Enterprise System, Ilmu Komputasi, Lama Mengajar, dan Membimbing Tugas Akhir.

2.5. Desain Arsitektural

Desain arsitektural merupakan desain yang digunakan untuk menggambarkan hubungan antara sub-sistem dan komponen yang berkaitan langsung dengan sistem. Desain arsitektur dapat dilihat pada Gambar 7.



Gambar 7. Desain Arsitektur Aplikasi Sipeka

Pada gambar di atas terlihat bahwa pengguna (mahasiswa atau koordinator TA) dapat menggunakan *smartphone* nya dengan menggunakan aplikasi Sipeka Dosen yang terhubung ke internet untuk bisa mengakses data pada aplikasi tersebut.

2.6. Metode Pengembangan Extreme Programming

Untuk mengembangkan suatu sistem perlu adanya metodologi yang dapat memandu dan memudahkan peneliti dalam menentukan langkah yang akan diambil berikutnya. Hal ini yang disebut dengan Metode Pengembangan sistem. Hal ini disebut juga sebagai *methodology software development*, yaitu suatu metode dalam pengembangan sistem sejak sistem didefinisikan, dirancang, diimplementasikan kemudian diuji kepada pengguna. Dalam metode *Agile*, interaksi dan personel lebih penting ketimbang proses dan alat, perangkat lunak yang berfungsi lebih penting daripada dokumentasi, kerja sama dengan client lebih penting daripada kontrak dan sikap terhadap perubahan lebih penting daripada mengikuti rencana [7].

Salah satu metode pengembangan sistem yang digunakan dalam penelitian ini adalah agile. Awalnya agile diciptakan untuk merespon yang lebih cepat jika terjadi permintaan atau penyesuaian dari klien. Metode yang sudah ada sebelumnya seperti antaranya adalah Extreme Programming (XP), Dynamic Systems Development Method (DSDM), Adaptive Software Development (ASD), Scrum Methodology sudah ada sejak lama namun metode ini terkesan terlalu dikontrol. Dengan demikian Agile model tentunya mempunyai kelebihan dibandingkan dengan metode lainnya [8].

2.7. Fuzzy Logic

Teori himpunan *Fuzzy Logic* pertama kali diperkenalkan oleh Lotti A. Zadeh pada tahun 1965. Pada *fuzzy logic* yang ditekankan adalah fungsi keanggotaan yang merepresentasikan derajat keanggotaan dari suatu objek dari atribut tertentu. Sehingga *fuzzy logic* adalah model pengembangan yang menangani masalah ketidakpastian, keraguan, ketidaktepatan, dan kebenaran yang bersifat Sebagian [9].

Fungsi keanggotaan tersebut digunakan untuk proses fuzzifikasi. Fuzzifikasi adalah suatu proses yang digunakan untuk mengubah himpunan *non-fuzzy* menjadi himpunan *fuzzy* sesuai dengan domain atau semesta pembicaraan input. Ada beberapa fungsi keanggotaan himpunan *fuzzy*, antara lain fungsi keanggotaan linier, fungsi keanggotaan segitiga, dan fungsi keanggotaan trapesium [10].

2.8. Multiple Attribute Decision Making (MADM)

Multiple Attribute Decision Making (MADM) adalah pendekatan klasik yang digunakan untuk menentukan penjumlahan terbobot pada setiap atribut [11]. Jadi MADM adalah metode pengambilan keputusan untuk menentukan alternatif yang terbaik dari sejumlah kriteria tertentu. Salah satu MADM yang digunakan dalam penelitian ini adalah *Simple Additive Weighting* (SAW).

Multiple Attribute Decision Making dalam prosesnya sebenarnya melakukan evaluasi terhadap sekumpulan atribut yang ada. Evaluasi yang dilakukan yaitu berupa nilai-nilai terbobot yang nantinya akan

dipilih dalam proses perankingan. MADM akan diakhiri dengan proses perankingan untuk mendapatkan nilai terbaik berdasarkan nilai kriteria yang sudah diberikan sebelumnya [12].

2.9. Simple Additive Weighting (SAW)

Simple Additive Weighting (SAW) adalah salah satu bagian dari metode MADM. Pertama kali yang harus dilakukan dalam metode SAW adalah menentukan bobot dari masing-masing kriteria yang diangkat dalam penelitian. Lalu mengumpulkan data atau alternatif yang disusun per kriteria. Data alternatif ini biasa disebut dengan matriks alternatif. Matriks alternatif ini harus diubah ke dalam nilai tertentu atau biasa disebut dengan proses normalisasi. Proses normalisasi meliputi penyesuaian kriteria sesuai dengan jenis atributnya apakah atribut biaya atau atribut keuntungan [13].

Persamaan untuk menghitung normalisasi dengan metode SAW dapat dilihat pada persamaan (1) berikut.

$$r_{ij} = \frac{x_{ij}}{\text{Max}_i x_{ij}}$$

jika j adalah atribut keuntungan (benefit)

$$r_{ij} = \frac{\text{Min}_i x_{ij}}{x_{ij}}$$

jika j adalah atribut biaya (cost)

(1)

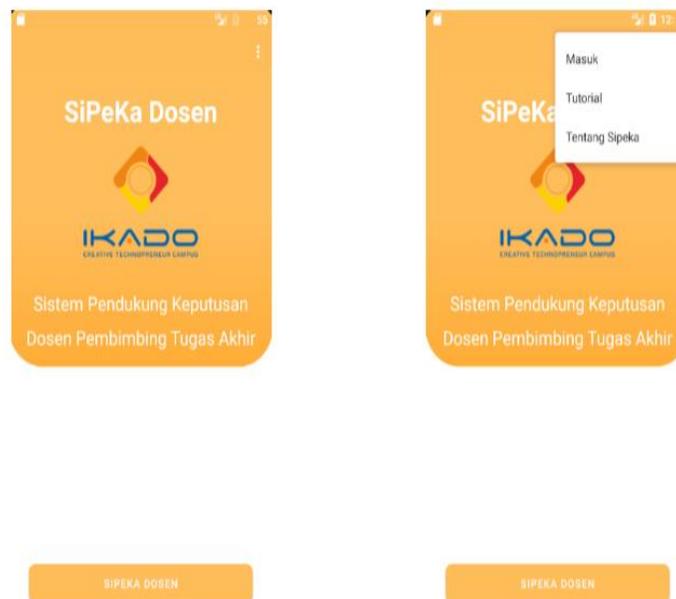
Keterangan:

- r_{ij} = Rating kinerja ternormalisasi
- X_{ij} = Nilai atribut yang dimiliki dari setiap kriteria.
- $\text{Max } X_{ij}$ = Nilai terbesar jika yang dicari adalah atribut keuntungan atau nilai tertinggi.
- $\text{Min } X_{ij}$ = Nilai min jika yang dicari adalah biaya atau nilai terendah.

3. Hasil dan Pembahasan

3.1. Hasil

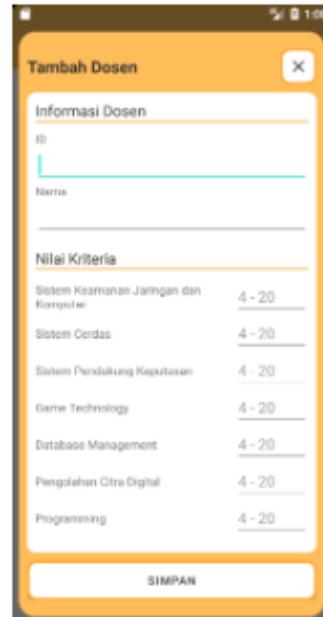
Berikut ini akan dijelaskan tampilan antar muka dari aplikasi SIPEKA yaitu gambar-gambar tampilan antar muka yang diterapkan pada aplikasi dengan penjelasan lengkap dari kegunaan atau fitur-fiturnya. Tampilan aplikasi SIPEKA dapat dilihat pada Gambar 8 sampai dengan Gambar 13.



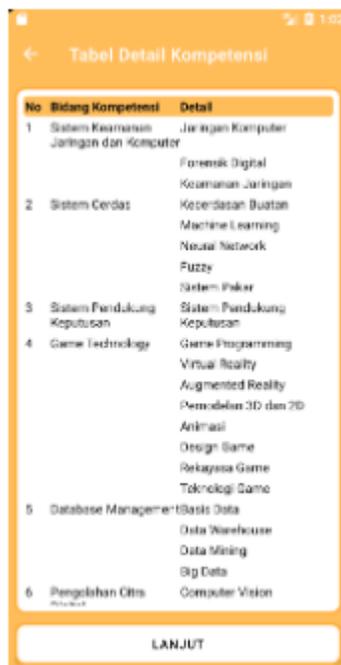
Gambar 8. Tampilan Menu Utama Aplikasi SIPEKA



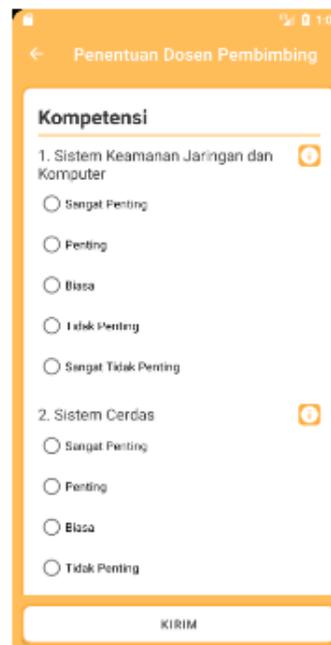
Gambar 9. Tampilan Login



Gambar 10. Tampilan Tambah Dosen



Gambar 11. Tampilan Detail Kriteria



Gambar 12. Tampilan Penentuan Dosen Pembimbing Untuk Mahasiswa



Gambar 13. Tampilan Penentuan Dosen Pembimbing Untuk Mahasiswa

3.2. Pembahasan

Dalam menentukan dosen pembimbing tugas akhir, dibutuhkan beberapa kriteria. Nantinya setiap alternatif akan dibandingkan berdasarkan kriteria-kriteria tersebut. Kriteria tersebut didapatkan dari referensi list daftar dosen pembimbing tugas akhir dan bidang kompetensi yang ada pada vcampus IKADO, konsultasi dengan dosen pembimbing, serta wawancara dengan dosen. Berikut adalah Tabel 2 adalah table kriteria dan Tabel 3 adalah tabel skala yang digunakan untuk menentukan dosen pembimbing tugas akhir:

Tabel 2. Tabel Kriteria

No	Kriteria	No	Kriteria
1	Sistem Keamanan Jaringan dan Komputer	8	Sistem Informasi
2	Sistem Cerdas	9	Bisnis Digital
3	Sistem Pendukung Keputusan	10	Rekayasa Perangkat Lunak
4	Game Technology	11	Enterprise System
5	Database Management	12	Ilmu Komputasi
6	Pengolahan Citra Digital	13	Lama Mengajar
7	Programming	14	Membimbing Tugas Akhir

Tabel 3. Tabel Skala

Keterangan	Skala
Sangat Tidak Menguasai	0 – 8
Tidak Menguasai	4 – 12
Cukup	8 – 16
Menguasai	12 – 20
Sangat Menguasai	16 – 20

Setelah menentukan kriteria untuk kebutuhan perhitungan *Simple Additive Weighting* kemudian selanjutnya dilakukan pengumpulan data terkait penguasaan dosen pembimbing dan juga data kriteria lain seperti lama mengajar dan membimbing tugas akhir. Pengumpulan data tersebut menggunakan media kuesioner yang dibuat dengan menggunakan metode *self-report* karena responden adalah orang yang paling tau tentang dirinya sendiri, berdasarkan pengalaman dosen sewaktu mengajar maupun membimbing tugas akhir, sehingga menghasilkan yang lebih akurat. Setelah data kriteria didapatkan kemudian selanjutnya menghitung data tersebut dengan menggunakan metode *fuzzy* dan SAW.

Berikut contoh perhitungan fuzzifikasi pada 12 bidang kompetensi yang dilakukan pada salah satu data dosen pembimbing. Misalnya didapatkan hasil dari kuesioner dengan total 15 maka akan diaplikasikan ke dalam *fuzzy membership function* dimana angka 15 sesuai dengan tabel skala maka skala penguasaan dosen akan masuk pada kategori cukup dan menguasai. Adanya 2 kategori ini yang membutuhkan penentuan keputusan lebih lanjut soal derajat keanggotaan penguasaan dosen sehingga dihitung menggunakan *fuzzy*. Nilai dosen akan dihitung menurut fungsi keanggotaan naik atau turun, setelah itu nilai terbesar yang akan digunakan. Dari nilai tersebut akan dijadikan ke bahasa linguistik, Pada Tabel 4 merupakan contoh hasil dari fuzzifikasi penguasaan dosen.

Tabel 4. Contoh Perhitungan Fuzzifikasi

Dosen	STM	TM	C	M	SM	Nilai	Fuzzifikasi	Hasil
17				0.75	0.25	0.75	Menguasai	0.75
8	0	1	0			1	Tidak Menguasai	0.25
5	0.75	0.25				0.75	Sangat Tidak Menguasai	0

Setelah menyelesaikan perhitungan fuzzifikasi maka dilanjutkan pada perhitungan dengan metode *Simple Additive Weighting*. Berikut contoh perhitungan pada tiga dosen, untuk perhitungan 12 kriteria bidang kompetensi, kriteria lama mengajar dan membimbing tugas akhir. Akan dilakukan normalisasi terlebih dahulu untuk kriteria lama mengajar dan membimbing tugas akhir.

Tabel 5. Contoh Data Kriteria

Dosen	Lama Mengajar	Membimbing Tugas Akhir
A1	17	50
A2	7	12
A3	5	10

Kemudian dari data kriteria lama mengajar dan membimbing tugas akhir dilakukan normalisasi dengan menggunakan persamaan 1 rumus normalisasi.

$$r_{ij} = \frac{x_{ij}}{\text{Max}_i x_{ij}} \quad \text{Persamaan (1)}$$

jika j adalah atribut keuntungan (benefit)

Keterangan:

- r_{ij} = Rating kinerja ternormalisasi
- X_{ij} = Nilai atribut yang dimiliki dari setiap kriteria.
- $\text{Max } X_{ij}$ = Nilai terbesar jika yang dicari adalah atribut keuntungan atau nilai tertinggi.

Normalisasi Lama Mengajar:

$$[A1]=17/17=1$$

$$[A2]=7/17=0.41$$

$$[A3]=5/17=0.29$$

Normalisasi Membimbing Tugas Akhir:

$$[A1]=50/50=1$$

$$[A2]=12/50=0.24$$

$$[A3]=10/50=0.2$$

Setelah terkumpul lengkap perhitungan normalisasi dari 12 kriteria maka selanjutnya dihitung dengan menggunakan Simple Additive Weighting seperti pada contoh matriks yang sudah tersusun hasil akhirnya pada Tabel 6.

Tabel 6. Matriks Kriteria Per Dosen

Kriteria	Dosen 1	Dosen 2	Dosen 3
1	0	0.75	0
2	0.25	0.25	0.5
3	0.5	0	0.25
4	0.25	0	0
5	0.5	0.25	0.5
6	0.25	0.25	0
7	0.5	0.75	0.5
8	0.5	0.25	0.75
9	0.5	0	0.25
10	0.5	0.25	0.75
11	0.25	0	0.25
12	0	0.25	0
13	1.00	0.41	0.29
14	1.00	0.24	0.20

Dari matriks diatas akan dilakukan perhitungan Simple Additive Weighting, untuk menghasilkan perangkaan dosen pembimbing tugas akhir.

$$A1 = (1) (0) + (0.5) (0.25) + (0) (0.5) + (0) (0.25) + (0.5) (0.5) + (0.5) (0.25) + (1) (0.5) + (0.5) (0.5) + (0) (0.5) + (0.5) (0.5) + (0) (0.25) + (0) (0) + (0.75) (1) + (0.75) (1) = 2.6125$$

$$A2 = (1) (0.75) + (0.5) (0.25) + (0) (0) + (0) (0) + (0.5) (0.25) + (0.5) (0.25) + (1) (0.75) + (0.5) (0.25) + (0) (0) + (0.5) (0.25) + (0) (0) + (0) (0.25) + (0.75) (0.41) + (0.75) (0.24) = 3.00$$

$$A3 = (1) (0) + (0.5) (0.5) + (0) (0.25) + (0) (0) + (0.5) (0.5) + (0.5) (0) + (1) (0.5) + (0.5) (0.75) + (0) (0.25) + (0.5) (0.75) + (0) (0.25) + (0) (0) + (0.75) (0.29) + (0.75) (0.20) = 2.1175$$

Jadi hasil dari perhitungan tersebut, menghasilkan alternatif terbaik dosen pembimbing tugas akhir yaitu A2 dengan hasil 3.00.

3.3. Hasil Pengujian

Untuk mendapatkan hasil yang optimal, aplikasi ini kemudian diuji menggunakan metode *Black Box testing* dan pengujian kegunaan dengan *Usability Testing*. *Black Box Testing* merupakan pengujian yang memeriksa hasil output berdasarkan kelas uji dan fungsional dari sistem yang ada [15]. Metode yang digunakan dalam pengujian aplikasi ini adalah dengan metode *Equivalence Partitioning*. Metode *Equivalence Partitioning* adalah metode pengujian dengan cara membagi data input menjadi sejumlah kelas data sesuai dengan domain input. Hasil uji coba dengan menggunakan *Black Box Testing*, sebagai berikut:

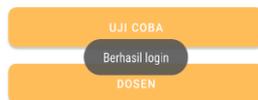
a) Hasil Pengujian Login

Pada Gambar 14, kondisi akhir halaman login berhasil dapat dilihat bahwa sistem dapat menampilkan halaman koordinator tugas akhir. Kemudian terdapat pesan “berhasil login” setelah melakukan login dan menekan tombol masuk. Koordinator tugas akhir berhasil masuk ke halaman selanjutnya yaitu halaman dosen.

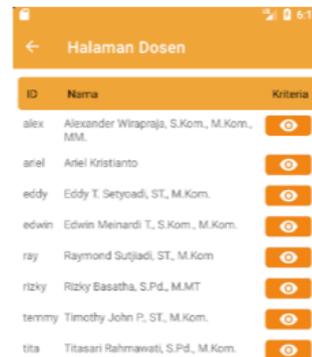
b) Hasil Pengujian Tambah Dosen

Pada Gambar 15, kondisi akhir halaman tambah dosen dapat dilihat bahwa sistem telah menambahkan data dosen dan pesan “input dosen berhasil”. Koordinator dapat melihat penambahan data baru yang berhasil ditambahkan pada halaman dosen.

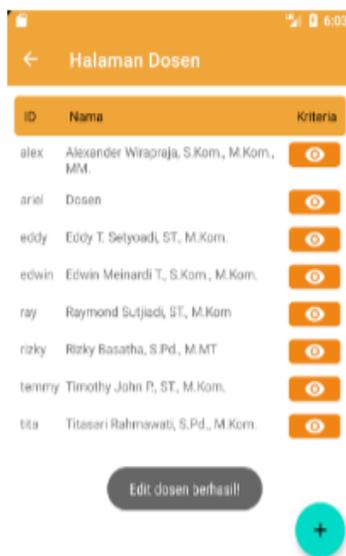
c) Pada gambar 16 kondisi akhir halaman edit dosen dapat dilihat bahwa sistem telah mengubah data dosen setelah koordinator melakukan edit dosen dan menekan tombol simpan. Koordinator dapat melihat perubahan data yang berhasil diubah pada halaman dosen dan memilih button lihat pada dosen yang diubah.



Gambar 14. Kondisi Akhir Pengujian Halaman Login Berhasil



Gambar 15. Kondisi Akhir Pengujian Halaman Tambah Dosen Berhasil



Gambar 16. Kondisi Akhir Pengujian Halaman Edit Dosen Berhasil

Sementara hasil pengujian dengan metode usability testing Usability adalah termasuk dalam kategori pengujian perangkat lunak khususnya untuk menguji seberapa mudah dan suatu aplikasi digunakan [16]. *Usability Testing* diukur dengan lima aspek yaitu:

- *Learnability* mengukur tingkat kemudahan dalam melakukan sejumlah *task* sederhana.
- *Efficiency* mengukur kecepatan pengguna dalam menyelesaikan sejumlah tujuan atau *task*.
- *Memorability* kemampuan pengguna dalam mempertahankan informasi dalam jangka waktu tertentu yang didukung dengan tata letak desain sistem yang konsisten. Sehingga pengguna yang tidak sering menggunakan sistem dapat menggunakan sistem Kembali dengan mudah dan cepat.
- *Errors* adalah seberapa banyak kesalahan yang dibuat pengguna dalam waktu tertentu. Kesalahan disini dihitung dengan menghitung jumlah kesalahan dalam menyelesaikan tugas.
- *Satisfaction* adalah berkaitan dengan persepsi pengguna, kepuasan, dan perasaan pengguna yang berinteraksi dengan sistem. *Satisfaction* dapat diukur dengan memberikan sejumlah pertanyaan secara subjektif kepada pengguna yang berinteraksi dengan sistem.

Berikut ini akan dijelaskan proses uji coba dengan menggunakan kuisioner. Kuisioner ini akan diberikan kepada 71 responden yang terdiri dari 70 mahasiswa dan 1 koordinator tugas akhir, berdasarkan jumlah populasi 85 orang. Serta menggunakan 4 aspek pada *usability*. Dari hasil perhitungan rata-rata dari 4 aspek *usability* didapatkan nilai-nilai pada Tabel 7. di bawah ini:

Tabel 7. Nilai Rata-rata Aspek 4 *Usability*

No	Variabel	Rata-Rata	Kategori
1	Memorability	3.944	Baik
2	Efficiency	3.958	Baik
3	Learnability	3.739	Baik
4	Satisfaction	3.669	Baik
	Rata-Rata	3.827	Baik

Dari hasil rata-rata 4 aspek diatas dapat disimpulkan bahwa aplikasi SIPEKA dikategorikan baik dengan skor 3.827.

4. Kesimpulan

Berdasarkan implementasi telah dilakukan terhadap aplikasi SIPEKA ini, diperoleh beberapa kesimpulan antara lain:

- Aplikasi SIPEKA dapat membantu mahasiswa di Institut Informatika Indonesia (IKADO) Surabaya untuk menentukan alternatif dosen pembimbing tugas akhir, serta membantu koordinator tugas akhir melakukan pengelolaan data dosen melalui fitur-fitur yang tersedia dalam aplikasi.
- Aplikasi ini menyajikan alternatif dosen pembimbing pertama dan kedua, beserta dengan lima peringkat teratas nama dosen yang dapat dipakai sebagai alternatif pembimbing.
- Dengan melakukan analisa perhitungan menggunakan *fuzzifikasi*, normalisasi, serta *simple additive weighting* (SAW) yang terintegrasi pada aplikasi Sipeka, dapat membantu koordinator tugas akhir dalam melakukan uji coba data-data dosen untuk menghasilkan alternatif nama-nama dosen pembimbing tugas akhir.
- Berdasarkan hasil uji coba yang dilakukan dengan 4 aspek dari metode *usability testing* dengan skor 3.827 dan kategori baik, membuktikan bahwa aplikasi Sipeka ini mudah untuk digunakan oleh mahasiswa dan koordinator tugas akhir.

Referensi

- [1] Dosen Pembimbing Akademik: Pengertian, Tugas dan Tips Menjadi Dosen PA, [Online]. Diakses dari <https://www.duniadosen.com/dosen-pembimbing-akademik/> . pada tanggal 02 Maret 2021 pukul 09.20 WIB.
- [2] aA. Evi Nur, W. M. Sri Umi, Keterlambatan Penyelesaian Skripsi mahasiswa Angkatan 2012 (Studi Kasus Di Jurusan Ekonomi Pembangunan Fakultas Ekonomi Universitas Negeri Malang. *Jurnal Pendidikan Ekonomi*, 10(1), 23-32, 2017
- [3] Amandus, H., Mawarni, D., Effendy, C., & Hasanbasri, M., 2018, Sulit Bertemu Dosen dan Merasa Tidak Memperoleh Masukan : Persepsi Mahasiswa Tentang Sosok Pembimbing Skripsi, *BKM Journal of Community Medicine and Public Health* Volume 34 No. 6.
- [4] L. Iwan, dkk. Sistem Pendukung Keputusan Dalam Menentukan Dosen Pembimbing Skripsi. *E-journal Teknik Informatika*. 9(1), 1-7, 2016
- [5] I. Anie Rose, S. Harry, Sistem Decision Support System Development For Determining The Preceptor Lectures In Proposing Internship Essay, Thesis And Final Project By Using SAW (Simple Additive Weighting) Method In Department Of Computer Science, Faculty Of Mathematic And Natural Science, Universitas Lampung. *Jurnal Komputasi*. 6(2), 76-96, 2018
- [6] Friyadi, Penerapan Metode Simple Additive Weight (SAW) Dalam Sistem Pendukung Keputusan Promosi Kenaikan Jabatan. *Jurnal Pilar Nusa Mandiri*. 12(1), 37-45, 2016
- [7] D. Sulisty, dkk. Metode Agile Dalam Perancangan Sitem Prediksi Prevelensi Stunting di Indonesia. *Jurnal Ilmiah NERO*. 5(2), 74-82, 2020
- [8] Widiarina, dkk, Metode Agile Pada Pembuatan Website Services and Sales Printer Ink. *Information Management For Educators And Professionals*. 4(2),143-152, 2020.
- [9] B. Haqi. *Aplikasi SPK Pemilihan Dosen Terbaik Metode Simple Additive Weighting*. Yogyakarta: Deepublish, 2019.
- [10] M. F. Al Haris, S. Kuswandi, *Kecerdasaan Buatan dan Aplikasinya*. Yogyakarta: CV. Absolute Media, 2019.
- [11] M. Ahsan, R. H. Susanti, R. N. Istiqomah, Multi-Attribute Decision Making Untuk Menentukan Multiple Intelligence Anak Menggunakan Metode Weighted Product. *Jurnal Informatika Merdeka Pasuruan*. 2(2), 24-34, 2017
- [12] R. Munadi, dkk. Penerapan Multiple Attribute Decision Making dengan Metode Simple Additive Weighting untuk Peningkatan Kerentanan Keamanan Website. *ELKOMIKA*. 6(2).194-206, 2018.
- [13] F. Sari, *Metode dalam Pengambilan Keputusan*. Yogyakarta: Deepublish, 2018.

- [14] A. R Pratama, Belajar UML – Use Case Diagram, *Codepolitan*, 2019, [Online]. Diakses dari <https://codepolitan.com/unified-modeling-language-uml> pada tanggal 22 Februari 2021 pukul 19.14 WIB.
- [15] A. Krismadi, Pengujian Black Box berbasis Equivalence Partitions pada Aplikasi Seleksi Promosi Kenaikan Jabatan. *Jurnal Teknologi Sistem Informasi dan Aplikasi*. 2(4), 155-161, 2019.
- [16] A. S, Penerapan Usability Testing Untuk Pengukuran Tingkat Kebergunaan Web Media Of Knowledge. *Jurnal Ilmiah Teknologi - Informasi dan Sains (TeknoIS)*. 8(1), 1-16, 2018.