

APLIKASI SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN GEDUNG SERBA GUNA DI KOTA BENGKULU DENGAN MENGGUNAKAN METODE SMART BERBASIS ANDROID

Nandik Sesnika¹, Desi Andreswari², Rusdi Efendi³

^{1,3}Program Studi Teknik Informatika, Fakultas Teknik, Universitas Bengkulu.
Jl. WR. Supratman Kandang Limun Bengkulu 38371A INDONESIA
(telp: 0736-341022; fax: 0736-341022)

¹sesnikanandikk@gmail.com

²deziandrez@yahoo.co.id

³r_efendi@yahoo.com

Abstrak: Pada penelitian ini dibangun sebuah aplikasi yang digunakan untuk memilih dan mencari informasi mengenai gedung serba guna di Kota Bengkulu. Aplikasi ini dibangun berbasis Android dengan menggunakan Bengkulu dengan menggunakan sebuah metode yang bernama *Simple Multi Attribute Rating Technique* (SMART), dan dibangun dengan menggunakan bahasa pemrograman JAVA dengan IDE ECLIPSE JUNO. Analisis perancangan sistem ini menggunakan *Unified Modeling Language* (UML). Dapat disimpulkan bahwa aplikasi ini dapat membantu para pengguna dalam memilih gedung serba guna yang ada di Kota Bengkulu dengan menggunakan metode SMART. Hasil pencarian yang ditampilkan berupa daftar nama gedung serba guna yang didalamnya terdapat berbagai informasi mengenai gedung-gedung serba guna tersebut.

Kata Kunci:Android, Gedung Serba Guna, Metode *Simple Multi Attribute Rating Technique* (SMART), *Eclipse IDE*, Java.

Abstract : *In this research, an application is developed as a mean to choose and search information about multipurpose building in Bengkulu City. This application was built on Android-based by using Simple Multi Attribute Rating Technique (SMART) Method and applied with JAVA programming language with Eclipse JUNO IDE. This system design analysis uses Unified Modeling Language (UML). It can be concluded that this application is able to help user for choose multipurpose building in Bengkulu City by using Simple Multi Attribute Rating Technique (SMART) Method. The displayed result of retrieval is list of multipurpose building name and information about multipurpose building. Keywords :Android, Digital Document, Simple Multi Attribute Rating Technique (SMART) Method, Eclipse IDE*

I. PENDAHULUAN

Perkembangan zaman yang semakin canggih, telah membawa banyak perubahan bagi kehidupan manusia, terutama dengan adanya teknologi. Teknologi terus dikembangkan dan semakin berkembang demi memenuhi tuntutan manusia terhadap berbagai kemudahan. Keberadaan teknologi dalam kehidupan kini telah menjadi suatu kebutuhan dan gaya hidup, salah satunya teknologi *mobile computing*.

Di Kota Bengkulu, terdapat banyak gedung-gedung serba guna. Dari survey awal yang dilakukan, di Kota Bengkulu terdapat 20 unit gedung serba guna. Gedung serba guna adalah gedung-gedung multifungsi, antara lain digunakan dalam acara rapat besar, seminar, pesta pernikahan, *workshop*, dan lain-lain. Banyaknya gedung serba guna di Kota Bengkulu yang tersebar secara tidak merata, lokasi yang berjauhan antar gedung serba guna yang ada dan disisi lain, tidak tersedianya sarana informasi yang memadai mengenai gedung-gedung tersebut menjadi kendala yang akan dihadapi oleh para konsumen ketika ingin melihat-lihat dan memilih gedung serba guna yang diinginkan. Para konsumen tersebut harus mendatangi satu per satu gedung-gedung serba guna dan pemilik gedung-gedung serba guna yang ada agar bisa mendapatkan informasi mengenai gedung-gedung serba guna tersebut.

Penulis memilih metode SMART sebagai metode yang digunakan yaitu karena SMART merupakan metode pengambilan keputusan yang fleksibel. SMART lebih banyak digunakan karena kesederhanaanya dalam merespon kebutuhan pembuat keputusan dan memilih android sebagai sistem operasi yang digunakan yaitu karena *mobile* dan tablet saat ini didominasi oleh sistem operasi android. Android juga merupakan suatu sistem *open source* yang sangat *powerfull*. Dunia sudah

menamainya sebagai bagian dari platform yang sangat populer.

Berdasarkan permasalahan yang telah diuraikan di atas, maka penulis tertarik untuk melakukan penelitian dengan judul “**Aplikasi Sistem Pendukung Keputusan Pemilihan Gedung Serba Guna di Kota Bengkulu dengan Menggunakan Metode SMART (*Simple Multi Attribute Rating Technique*) Berbasis Android**”.

II. LANDASAN TEORI

A. *Sistem Pendukung Keputusan/Decision Support System*

Dokumen Decision Support System (DSS) merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semiterstruktur dan situasi yang tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat [1].

Tujuan dari DSS menurut Turban dalam (kusrini, 2007) adalah :

- Membantu manajer dalam pengambilan keputusan atas masalah semiterstruktur.
 - Memberikan dukungan atas pertimbangan manajer dan bukannya dimaksudkan untuk menggantikan fungsi manajer.
 - Meningkatkan efektivitas keputusan yang diambil manajer lebih dari perbaikan efisiensinya.
 - Kecepatan Komputasi. Komputer memungkinkan para pengambil keputusan untuk melakukan banyak komputasi secara cepat dengan biaya yang rendah.
 - Peningkatan produktivitas, membangun suatu kelompok pengambilan keputusan, terutama para pakar, biaya sangat mahal.
 - Dukungan kualitas
-

- g. Berdaya saing
- h. Mengatasi keterbatasan kognitif dalam pemrosesan dan penyimpanan.

B. Metode Simple Multi Attribute Rating Technique (SMART)

Menurut (Kustiyahningsih, et al., 2010), SMART merupakan metode dalam pengambilan keputusan multiatribut. Teknik pengambilan keputusan multiatribut ini digunakan untuk mendukung pembuat keputusan dalam memilih beberapa alternatif. Setiap pembuat keputusan harus memiliki sebuah alternatif yang sesuai dengan tujuan yang dirumuskan. Setiap alternatif terdiri dari sekumpulan atribut dan setiap atribut mempunyai nilai-nilai. Nilai ini dirata-rata dengan skala tertentu. Setiap atribut mempunyai bobot yang menggambarkan seberapa penting suatu atribut dibandingkan dengan atribut lain. Pembobotan dan pemberian peringkat ini digunakan untuk menilai setiap alternatif agar diperoleh alternatif terbaik [2].

Model yang digunakan dalam SMART yaitu :

$$u(a_i) = \sum_{j=1}^m w_j u_i(a_i) \quad i=1,2,\dots,m$$

Keterangan:

w_j = nilai pembobotan kriteria ke-j

$u_i(a_i)$ = nilai utility kriteria ke-i

Adapun Teknik dalam metode SMART yaitu :

1. Menentukan jumlah kriteria
2. Menentukan persentase bobot kriteria. (Untuk penentuan bobot kriteria, menggunakan rumus dari pembobotan ROC (*Rank Order Centroid*), pemberian bobot pada setiap kriteria dilakukan sesuai dengan ranking yang dinilai berdasarkan

tingkat prioritas. Biasanya dibentuk dengan pernyataan “Kriteria 1 lebih penting dari kriteria 2, yang lebih penting dari kriteria 3” dan seterusnya hingga kriteria ke n, ditulis , kemudian normalisasi.

$$\text{Normalisasi} = \frac{w_j}{\sum w_j}$$

....(2.2)

Keterangan :

w_j : bobot suatu kriteria

$\sum w_j$: total bobot semua kriteria

3. Memberikan nilai kriteria untuk setiap alternatif.
4. Hitung nilai utility untuk setiap subkriteria masing-masing.

$$u_i(a_i) = 100\% \frac{(C_{max} - C_{out\ i})}{(C_{max} - C_{min})}$$

...(2.3)

Keterangan :

$u_i(a_i)$: nilai utility kriteria ke-1 untuk kriteria ke-i

C_{max} : nilai kriteria maksimal

C_{min} : nilai kriteria minicmal

$C_{out\ i}$: nilai kriteria ke-i

Tetapi dalam kasus ini, pemberian nilai utility juga dihitung dengan menggunakan pembobotan ROC (*Rank Order Centroid*).

5. Hitung nilai akhir masing-masing dengan menggunakan rumus dari metode SMART.

$$u(a_i) = \sum_{j=1}^m w_j u_i(a_i) \quad i=1,2,\dots,m$$

C. Rank Order Centroid

Algoritma ROC didasarkan pada tingkat kepentingan atau prioritas dari kriteria. Menurut

Jeffreys dan Cockfield dalam (Rahma, 2013), teknik ROC memberikan bobot pada setiap kriteria sesuai dengan ranking yang dinilai berdasarkan tingkat prioritas. Biasanya dibentuk dengan pernyataan “Kriteria 1 lebih penting dari kriteria 2, yang lebih penting dari kriteria 3” dan seterusnya hingga kriteria ke n ditulis. Untuk menentukan bobotnya, diberikan aturan yang sama yaitu dimana merupakan bobot untuk kriteria. Atau dapat dijelaskan sebagai berikut[3]:

Jika

$$Cr1 \geq Cr2 \geq Cr3 \geq \dots \geq Crn$$

Maka

$$W1 \geq W2 \geq W3 \geq \dots \geq Wn,$$

Selanjutnya, jika k merupakan banyaknya kriteria, maka

$$\begin{aligned} W1 &= \frac{1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{k}}{k} \\ W2 &= \frac{0 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{k}}{k} \\ W3 &= \frac{0 + 0 + \frac{1}{3} + \dots + \frac{1}{k}}{k} \\ W_k &= \frac{0 + \dots + 0 + \frac{1}{k}}{k} \dots (2.5) \end{aligned}$$

Secara umum pembobotan ROC, dapat dirumuskan sebagai berikut,

$$W_k = \frac{1}{k} \sum_{i=1}^k \left(\frac{1}{i} \right) \dots (2.6)$$

D. Sistem Operasi Android

Sistem *Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. *Android* menyediakan *platform* yang terbuka bagi para pengembang untuk menciptakan aplikasi sendiri. Awalnya, Google Inc. membeli *Android Inc* yang merupakan pendatang baru yang membuat piranti lunak untuk ponsel atau *smartphone*.

Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan piranti keras, piranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile dan Nvidia (Safaat, 2012) [4].

Perangkat Lunak Yang Mendukung, antara lain :

1) *Java*: Sejarah *Java* berawal tahun 1991 ketika perusahaan Sun Microsystem memulai *Green Project*, yakni proyek penelitian untuk membuat bahasa yang akan digunakan pada *chip-chip embedded* untuk *device intelligent customer electronic*. Bahasa tersebut haruslah bersifat multi platform, tidak tergantung pada vendor yang memmanufaktur chip tersebut. Dalam penelitiannya, proyek *Green* berhasil membuat *prototype* semacam PDA (*Personal Data Assistance*) yang dapat berkomunikasi antara satu dengan yang lain yang diberi nama *Star 7*. Ide berawal untuk membuat sistem operasi bagi *Star 7* berbasis C dan C++. Setelah berjalan beberapa lama, James Gosling, salah satu seorang anggota tim merasa kurang puas dengan beberapa karakteristik dari kedua bahasa tersebut, dan berusaha mengembangkan bahasa lain. Bahasa tersebut kemudian dinamakan *Oak*, dan belakangan *Oak* beralih nama menjadi *Java* [4].

2) *Android SDK*: *Android SDK* adalah *tools API (Application Programming Interface)* yang diperlukan untuk mulai mengembangkan aplikasi pada *platform Android* menggunakan bahasa pemrograman *Java*. Saat ini disediakan *Android SDK* sebagai alat bantu dan *API* untuk mulai mengembangkan aplikasi pada *platform Android* menggunakan bahasa pemrograman *Java*[4].

3) *Eclipse IDE*: Sebuah pengembangan lingkungan yang teintegrasi yang dapat membantu untuk membangun aplikasi-aplikasi dalam banyak bahasa. Fitur yang terdapat dalam Eclipse IDE ini adalah sebagai berikut[5] :

1. *Package Explorer*.

Package Explorer berfungsi untuk menyimpan *file* dari proyek yang dibuat. *File* yang telah dibuat akan tersimpan menurut foldernya masing-masing

2. *Pallette*, berfungsi untuk menyimpan berbagai macam *widgets* atau fitur pada aplikasi Android nantinya.

3. *Graphical Layout*/desain, berfungsi untuk tempat rancangan desain aplikasi yang akan dibuat

4. *Outline*, berfungsi untuk menampilkan skema dari rangkaian fitur yang telah ditambahkan ke dalam proyek yang dibuat.

5. *Properties*, berfungsi sebagai tempat untuk fitur pendukung dari setiap proyek yang dibuat.

E. *Unified Modeling Language (UML)*

UML (Unified Modeling Language) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (*Object-Oriented*). UML terdiri atas pengelompokan diagram-diagram sistem. Diagram adalah yang menggambarkan permasalahan maupun solusi dari permasalahan suatu model. Salah satu cara untuk mengatur diagram UML adalah dengan menggunakan *view*. *View* adalah

kumpulan dari diagram yang menggambarkan aspek yang sama dari proyek yang terdiri dari *Static View*, *Dynamic View*, dan *Functional View* (Pender, 2002) [5].

III. METODOLOGI PENELITIAN

A. *Jenis Penelitian*

Jenis penelitian yang akan dilakukan adalah penelitian terapan (*applied research*). Merancang dan membangun sebuah aplikasi untuk mendukung keputusan dalam memilih gedung serba guna yang diinginkan dengan menggunakan metode SMART (*Simple Multi Attribute Rating Technique*) .

B. *Teknik Pengumpulan Data*

Teknik pengumpulan data yang digunakan pada penelitian ini adalah sebagai berikut:

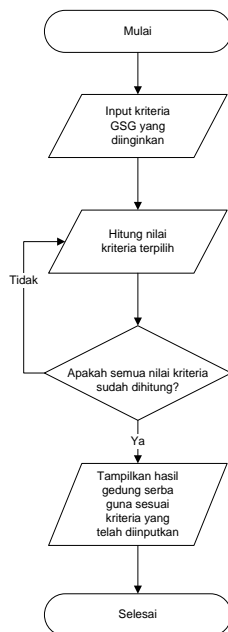
1) *Pustaka*: Teknik pengumpulan data yang diperoleh dari buku-buku dan/ atau jurnal dalam pencarian referensi terkait pengumpulan data maupun perancangan aplikasi yang akan dibangun.

2) *Teknik Observasi*: Suatu cara pengumpulan data dengan mengadakan pengamatan langsung terhadap suatu obyek dalam suatu periode tertentu dan mengadakan pencatatan secara sistematis tentang hal-hal tertentu yang diamati. Banyaknya periode observasi yang perlu dilakukan dan panjangnya waktu pada setiap periode observasi tergantung kepada jenis data yang dikumpulkan berdasarkan pola yang telah diberikan.

IV. ANALISIS DAN PERANCANGAN

A. Cara Kerja sistem

Secara garis besar cara kerja sistem yang dibangun ditampilkan dalam Gambar 1 berikut.



Gambar 1. Diagram Alir Antarmuka Pengguna dengan Sistem

Keterangan Diagram Alir :

1. Aplikasi dimulai dengan cara pengguna terlebih dahulu memilih menu pemilihan gedung serba guna.
2. Setelah pengguna memilih menu pemilihan gedung serba guna, kemudian pengguna akan memilih kriteria gedung serba guna yang diinginkan dengan mengklik pilihan sub kriteria yang ada (sebanyak 8 kriteria yang dipilih secara berurutan).
3. Setelah semua kriteria dipilih (hingga kriteria ke-8) maka pengguna akan mengklik button lanjut, setelah itu akan muncul sebuah halaman yang berisi tentang semua kriteria

yang telah dipilih tadi beserta nilai utility disetiap subkriterianya. Pada bagian bawah halaman ini, akan ada 2 tombol yang bisa di klik, yang pertama tombol daftar gedung yang tidak memiliki nilai, dan daftar gedung yang memiliki nilai. Setiap nama gedung tersebut ketika di klik akan menampilkan berbagai macam informasi mengenai gedung serba guna tersebut.

4. Pada menu “Petunjuk Aplikasi”, akan ditampilkan beberapa informasi mengenai tata cara penggunaan aplikasi.
5. Apabila pengguna ingin membuka menu “Tentang Aplikasi dan Programmer” maka sistem akan menampilkan informasi tentang pembuat atau *programmer* dari aplikasi yang digunakan. Jika tidak maka kerja sistem selesai.
6. Apabila pengguna tidak ingin membuka menu aplikasi dan tidak memilih menu, maka aplikasi akan selesai.

B. Perancangan Model UML (Unified Modeling Language)

Perancangan aplikasi yang mengimplementasikan metode *Simple Multi Attribute Rating Technique* (SMART) pada aplikasi system pendukung keputusan pemilihan gedung serba guna berbasis Android menggunakan UML (*Unified Modeling Language*), seperti berikut :

1) Use Case Diagram:



Gambar 2. Usecase Diagram

Pada aplikasi pencarian dokumen *digital* yang akan dibangun, terdapat interaksi antara pengguna dan sistem. Pada aplikasi ini, pencarian dokumen *digital* membutuhkan satu pengguna untuk menjalankan aplikasi tersebut.

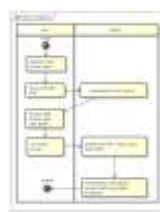
Tabel 1 Interaksi Pengguna dan interaksi sistem.

Aktor	Interaksi dengan sistem
Pengguna (<i>User</i>)	<ol style="list-style-type: none"> 1. Menu pemilihan gedung serba guna 2. Membaca petunjuk sistem 3. Memilih kriteria Gedung serba guna yang diinginkan. 4. Melihat informasi <i>output</i> yang berupa alternatif (gedung serba guna) yang sesuai dengan kriteria-kriteria yang telah diinputkan sebelumnya. 5. Melihat informasi tentang Sistem

2. *Activity Diagram*: Pada UML (*Unified Modeling Language*), *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses. *Activity Diagram* pada sistem ini terbagi menjadi 3 diagram, yaitu:

a. *Activity Diagram* Pemilihan Gedung Serba Guna

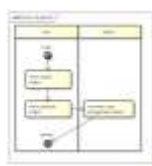
Activity Diagram pada pilihan menu “Pemilihan Gedung Serba Guna” ditunjukkan dalam Gambar 3 berikut.

Gambar 3. *Activity Diagram* Pemilihan Gedung Serba Guna

Pada Gambar .3 menunjukkan keseluruhan dari kerja sistem. Pada aplikasi, pencarian diawali dari pengguna terlebih dahulu memilih menu pemilihan gedung serba guna, kemudian sistem akan menampilkan formk gedung serba guna, setelah semua kriteria dipilih , klik tombol proses untuk melihat hasil yaitu berupa daftar nama gedung serba guna terpilih yang sesuai dengan criteria yang dmasukkan tadi. Setiap nama gedung yang tampil, jika diklik akan menampilkan informasi gedung beserta peta dan foto gedung tersebut. Setelah hasil ditampilkan, maka proses pencarian selesai dilakukan.

b. *Activity Diagram* Petunjuk Aplikasi

Activity Diagram pada pilihan menu “Petunjuk Aplikasi” ditunjukkan dalam Gambar 4 berikut.

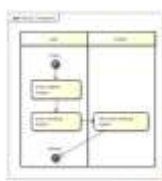


Gambar 4. *Activity Diagram* Petunjuk Aplikasi

Pada Gambar 4 merupakan *Activity Diagram* pada menu Petunjuk Aplikasi. Diawali dari pengguna memilih menu pilihan “petunjuk aplikasi” dimana pengguna dapat mengetahui tata cara atau penggunaan aplikasi tersebut. Setelah informasi petunjuk aplikasi ditampilkan, maka proses selesai.

c. *Activity Diagram* Tentang Aplikasi

Activity Diagram pada pilihan menu “Tentang Aplikasi dan Programmer” ditunjukkan dalam Gambar 5 berikut.

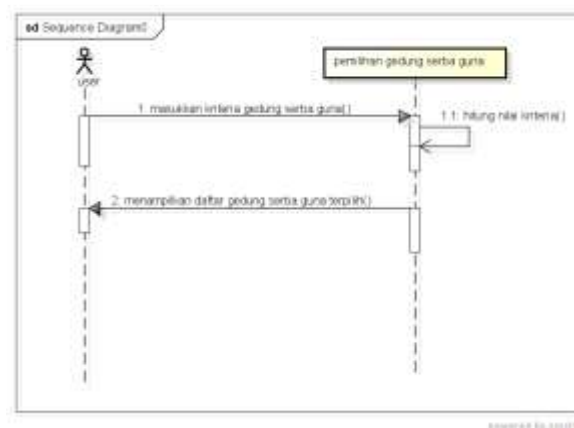


Gambar 5. *Activity Diagram* Tentang aplikasi dan Programmer

Pada Gambar 5 merupakan *Activity Diagram* pada menu Tentang Aplikasi dan Programmer. Diawali dari pengguna memilih menu pilihan “tentang Aplikasi dan programmer” dimana pengguna mendapat informasi mengenai pembuat atau programmer aplikasi tersebut. Setelah informasi mengenai tentang aplikasi ditampilkan, maka proses selesai.

3) *Sequence Diagram*: Merupakan gambaran *Sequence Diagram* dari aplikasi system pendukung keputusan pemilihan gedung serba guna berbasis Android yang dijabarkan dalam Gambar 6 berikut :

a. *Sequence Diagram* Pemilihan Gedung Serba Guna

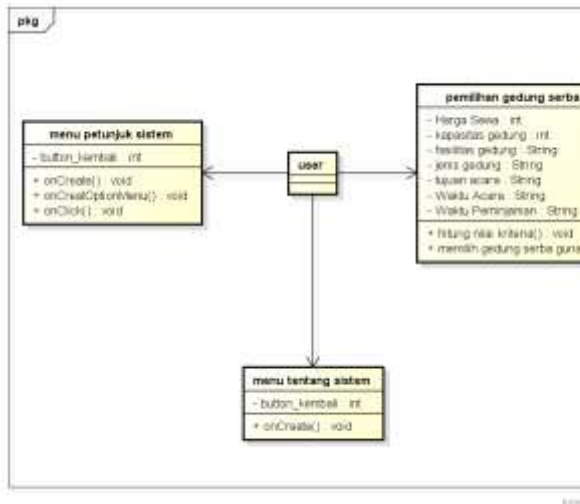


Gambar 6. *Sequence Diagram* Pencarian Dokumen

Pada diagram diatas, adalah interaksi antara pengguna dengan sistem. Pengguna terlebih dahulu memasukkan criteria Gedung Serba Guna yang diinginkan. Kemudian Sistem akan menampilkan hasil dari criteria yang telah diinputkan sebelumnya lengkap beserta nilainya, setelah itu system akan menampilkan daftar gedung serba guna yang sesuai dengan kriteria yang telah dipilih.

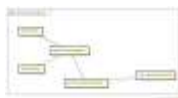
4) *Class Diagram*: *Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Pada sistem ini disetiap menu

yang ditampilkan memiliki operasi dan atribut-atribut yang mengikuti. Adapun *class diagram* untuk aplikasi pencarian dokumen *digital* dijabarkan dalam Gambar 7 berikut.



Gambar 7. Class Diagram Pemilihan Gedung Serba Guna

5) Object Diagram: Object *diagram* sangat mirip dengan *class diagram*. Perbedaan utamanya adalah bahwa diagram objek menggambarkan objek dan hubungannya. Pada gambar 8 dapat dilihat bahwa halaman utama pengguna mempunyai hubungan terhadap tiga objek yaitu tentang sistem, petunjuk sistem, pemilihan gedung serba guna. Pada objek pemilihan gedung serba guna terdapat hubungan dengan satu objek yaitu kriteria gedung.



Gambar 8. Object Diagram Pemilihan Gedung Serba Guna

6) Statechart Diagram: State chart diagram menghubungkan *event-event* dan *state-state*. Ketika suatu *event* diterima, *state* berikutnya bergantung pada *state* yang sekarang ada. Gambar

9 dibawah ini menggambarkan diagram state untuk sistem pemilihan gedung serba guna di Kota Bengkulu. Dapat dilihat pada Gambar 9 bahwa pada pemilihan gedung serba guna pengguna harus menginputkan kriteria, kemudian sistem akan menghitung nilai utility kriteria terpilih. Lalu akan diproses gedung serba guna sesuai dengan kriteria yang telah dipilih. Dan akan tampil hasil data gedung serba guna yang terbaik sesuai kebutuhan *user*.



Gambar 9. Statechart Diagram Pemilihan Gedung Serba Guna

7) Collaboration Diagram: Pada *collaboration diagram* setiap *event* mengirimkan informasi dari suatu objek ke objek yang lainnya. Pengguna memilih menu pemilihan gedung serba guna, lalu pengguna menginputkan kriteria yang diinginkan. Setelah itu proses gedung serba guna sesuai dengan kriteria yang telah diinputkan. Kemudian akan tampil gedung serba guna yang baik sesuai kebutuhan user. Gambar 10 adalah *collaboration diagram* sistem pemilihan gedung serba guna di Kota Bengkulu.



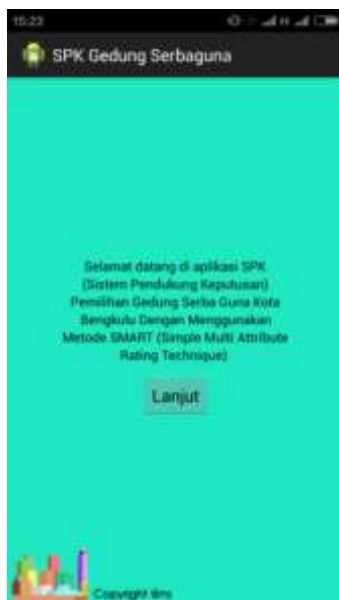
Gambar 10. Collaboration Diagram Pemilihan Gedung Serba Guna

V. PEMBAHASAN

A. Implementasi Antarmuka

Pada tahapan implementasi antarmuka ini, sistem akan diimplementasikan menggunakan bahasa pemrograman Java J2ME dengan menggunakan *Eclipse IDE*. Berikut tampilan hasil implementasi sistem:

1) *Halaman Depan Aplikasi*: Halaman depan aplikasi ini merupakan halaman aplikasi yang akan tampil pertama kali pada saat aplikasi dijalankan ketika memilih ikon aplikasi. Pada halaman depan aplikasi ini menampilkan nama dari aplikasi, logo folder dan *progress bar*.



Gambar 11. Tampilan Halaman Depan Aplikasi di Smartphone

2) *Halaman Menu Utama Aplikasi*: Setelah tampilan halaman depan selesai, maka tampilan selanjutnya adalah halaman menu utama aplikasi. Pada halaman menu utama ini terdapat empat pilihan menu, yaitu menu pemilihan gedung serba guna, menu petunjuk aplikasi dan menu tentang aplikasi dan programmer.



Gambar 12. Tampilan Menu Utama Di Smartphone

3) *Halaman Menu Pemilihan Gedung Serba Guna*: Halaman pemilihan gedung serba guna merupakan pemilihan utama yang terdapat dalam aplikasi ini. Menu pemilihan gedung serba guna merupakan menu pemilihan yang menggunakan Metode *Simple Multi Attribute Rating Technique* (SMART) dan menggunakan beberapa pilihan kriteria untuk pemilihan gedung serba guna ini, jika kita memilih menu pemilihan gedung serba guna, maka akan muncul sebuah tampilan yang berisikan pertanyaan untuk kriteria pertama :



Gambar 13. Tampilan Halaman Kriteria Harga Sewa

Pada Gambar 13 terdapat tiga pilihan sub kriteria, yaitu subkriteria dengan rentang harga sewa dari Rp. 600.000 – Rp. 4.000.000, yang kedua > Rp. 4.000.000 – Rp. 8.000.000, dan yang ketiga > Rp. 8.000.000,- . Ketika salah satu sub kriteria telah dipilih, kemudian pengguna mengklik tombol lanjut, maka akan muncul halaman pertanyaan kriteria kedua yang dapat dilihat seperti gambar 14 dibawah ini :



Gambar 14. Tampilan Halaman Kriteria Kapasitas Gedung

Pada gambar 14, kriteria kapasitas gedung juga memiliki 3 pilihan subkriteria yaitu : > 800 orang, >400-800 orang, dan

0-400 orang. Setelah pengguna memilih satu satu dari 3 subkriteria tersebut, langkah selanjutnya mengklik tombol lanjut, dan akan muncul halaman pertanyaan kriteria yang berikutnya. Dibawah ini merupakan tampilan halaman pertanyaan kriteria yang tersedia :



Gambar 15. Tampilan Halaman Kriteria Fasilitas Gedung



Gambar 16. Tampilan Halaman Kriteria Jenis Gedung



Gambar 17. Tampilan Halaman Kriteria Tujuan Acara



Gambar 19. Tampilan Halaman Kriteria Waktu Peminjaman



Gambar 18. Tampilan Halaman Kriteria Waktu Acara



Gambar 20. Tampilan Halaman Kriteria Lahan Parkir

Jika pengguna telah memilih semua kriteria yang telah disediakan, maka akan muncul sebuah halaman yang berisikan semua kriteria yang telah dipilih tadi, beserta nilai utility nya. Tampilan

halaman tersebut dapat kita lihat seperti gambar 21 dibawah ini :



Gambar 21. Tampilan Halaman Kriteria terpilih

Ketika tombol Lihat Hasil diklik, maka akan muncul sebuah tampilan halaman alternatif yaitu berupa daftar nama gedung serba guna terpilih beserta nilai gedung serba guna tersebut. Tampilan halaman tersebut dapat kita lihat seperti gambar 22 dibawah ini :



Gambar 22. Tampilan Halaman alternatif terpilih

Pada Gambar diatas, terdapat nilai dan nama gedung. Nilai yang tertera yaitu merupakan nilai akhir dari suatu gedung yang didapatkan dengan mengalikan nilai utility dan nilai bobot suatu kriterianya. Jika nama gedung serba guna tersebut diklik, maka akan muncul sebuah halaman yang berisikan informasi mengenai gedung serba guna tersebut. Dapat kita lihat pada gambar 23 dibawah ini :



Gambar 23. Tampilan Halaman informasi gedung

4) *Halaman Petunjuk Aplikasi*: Halaman petunjuk aplikasi merupakan menu yang terdapat dalam aplikasi pencarian dokumen ini. Halaman petunjuk aplikasi ini berisikan tata cara penggunaan dari aplikasi pemilihan gedung serba guna tersebut. Halaman petunjuk aplikasi ini dapat membantu pengguna dalam memahami dalam menggunakan menu-menu yang terdapat di aplikasi ini.



Gambar 24. Tampilan Halaman Petunjuk Aplikasi di Smartphone

5) *Halaman Tentang Aplikasi*: Halaman tentang aplikasi merupakan halaman yang berisikan informasi mengenai aplikasi pemilihan gedung serba dan pembuat aplikasi tersebut.



Gambar 25. Tampilan Halaman Tentang Aplikasi

B. Pengujian Sistem Secara Manual

Tabel dibawah merupakan tabel yang berisi tahap untuk pencarian nilai utility dari suatu subkriteria. Setelah subkriteria di dapat, maka akan lanjut ke tahap berikutnya yaitu perhitungan nilai akhir dari suatu subkriteria yang dapat dilihat pada tabel dibawah ini :

KRITERIA	Nilai Utility	Bobot	Nilai Akhir
BANGSA SERBA			
Ba. 000.000 - Rp. 0.000.000	$(1 + 0,5 - 0,5) / 2 = 0,5$	0,301	0,1505
0 Rp. 1.000.000 - Rp. 8.000.000	$(0,5 + 0,5) / 2 = 0,5$	0,499	0,2495
2 Rp. 8.000.000	$(1,5) / 3 = 0,5$	0,201	0,1005
KAPASITAS GEDUNG			
0-100	$(1 + 0,5 - 0,5) / 2 = 0,5$	0,301	0,1505
100-200	$(0,5 + 0,5) / 2 = 0,5$	0,499	0,2495
200-300	$(1,5) / 3 = 0,5$	0,201	0,1005
FASILITAS GEDUNG			
0-10	$(1 + 0,5 - 0,5) / 2 = 0,5$	0,301	0,1505
10-20	$(0,5 + 0,5) / 2 = 0,5$	0,499	0,2495
20-30	$(1,5) / 3 = 0,5$	0,201	0,1005
WAKTU ACARA			
0-10	$(1 + 0,5 - 0,5) / 2 = 0,5$	0,301	0,1505
10-20	$(0,5 + 0,5) / 2 = 0,5$	0,499	0,2495
20-30	$(1,5) / 3 = 0,5$	0,201	0,1005
WAKTU PEMERIKSAAN			
0-10	$(1 + 0,5 - 0,5) / 2 = 0,5$	0,301	0,1505
10-20	$(0,5 + 0,5) / 2 = 0,5$	0,499	0,2495
20-30	$(1,5) / 3 = 0,5$	0,201	0,1005
KAPASITAS PARKIR			
0-10	$(1 + 0,5 - 0,5) / 2 = 0,5$	0,301	0,1505
10-20	$(0,5 + 0,5) / 2 = 0,5$	0,499	0,2495
20-30	$(1,5) / 3 = 0,5$	0,201	0,1005

No	Nilai Kriteria	Nilai Utility : $100\% \frac{(C_{max} - C_{out})}{(C_{max} - C_{min})}$
1	0,750	$100\% \frac{(0,750 - 0,750)}{(0,750 - 0,0625)} = 0$
2	0,611	$100\% \frac{(0,750 - 0,611)}{(0,750 - 0,0625)} = 0,202$
3	0,520	$100\% \frac{(0,750 - 0,520)}{(0,750 - 0,0625)} = 0,334$
4	0,278	$100\% \frac{(0,750 - 0,278)}{(0,750 - 0,0625)} = 0,686$
5	0,270	$100\% \frac{(0,750 - 0,270)}{(0,750 - 0,0625)} = 0,698$
6	0,250	$100\% \frac{(0,750 - 0,250)}{(0,750 - 0,0625)} = 0,727$
7	0,145	$100\% \frac{(0,750 - 0,145)}{(0,750 - 0,0625)} = 0,880$
8	0,110	$100\% \frac{(0,750 - 0,110)}{(0,750 - 0,0625)} = 0,931$
9	0,0625	$100\% \frac{(0,750 - 0,0625)}{(0,750 - 0,0625)} = 1$

Ketika semu nilai akhir dari suatu subkriteria telah dihitung, maka akan masuk ketahap perhitungan terakhir dalam metode SMART yaitu pencarian jumlah keseluruhan subkriteria untuk suatu gedung agar menghasilkan nilai akhir dari gedung tersebut dengan rumus :

$$u(a_i) = \sum_{j=1}^m w_j u_i(a_i) \quad i=1,2,\dots,m$$

Keterangan:

w_j = nilai pembobotan kriteria ke-j

$u_i(a_i)$ = nilai utility kriteria ke-i

VI. KESIMPULAN

Berdasarkan analisa perancangan sistem, implementasi, dan pengujian sistem, maka dapat disimpulkan bahwa penelitian ini telah berhasil dibangun dengan model perancangan *Unified Modelling Language* atau UML dan menggunakan bahasa pemrograman Java dengan mengimplementasikan metode SMART yang dijalankan di Android. Sistem ini dapat memberikan kemudahan kepada pengguna dalam memilih gedung serba guna. Dalam pengujian kemudahan sistem, didapatkan hasil penilaian, yaitu kemudahan dalam penggunaan sistem berfungsi dengan baik, kemudahan dalam penginputan data sangat baik, dan kemudahan dalam mengakses informasi sangat baik, fungsi-fungsi dalam sistem berfungsi dengan sangat baik, antarmuka dan pengaksesan sistem sangat baik, dan sistem bermanfaat untuk pengguna.

VII. SARAN

Berdasarkan Berdasarkan hasil penelitian, pengujian, implementasi serta pembahasan mengenai Aplikasi Sistem Pendukung Keputusan Pemilihan Gedung Serba Guna Berbasis Android (Studi Kasus : GSG Kota Bengkulu), maka untuk pengembangan penelitian selanjutnya penulis menyarankan sebagai berikut:

1. Aplikasi Sistem ini dapat terus dikembangkan lebih lanjut dalam hal metode yang digunakan, kedepannya diharapkan untuk dapat menggunakan metode selain metode SMART, seperti metode SMARTER, Electre,

dan lain-lain dalam pemilihan gedung serba guna ini.

2. Sistem ini dapat dilakukan pengembangan sistem dalam hal perangkat, kedepannya diharapkan dapat digunakan untuk perangkat online (*web*).
3. Mengembangkan aplikasi yang dapat digunakan pada perangkat bergerak lainnya yang mempunyai sistem operasi tersendiri seperti *iphone*.

REFERENSI

- [1] Kusrini. (2007). Konsep dan Aplikasi Sistem Pendukung Keputusan. Yogyakarta : ANDI.
- [2] Kustiyahningsih, Y., Anamisa, D. R., & Syafa'ah, N. (2010). Sistem Pendukung Keputusan Untuk Menentukan Jurusan Pada Siswa SMA Menggunakan Metode KNN dan SMART. 1-10.
- [3] Rahma, Afiefah. 2013. Sistem Pendukung Keputusan Seleksi Mahasiswa Menggunakan Metode SMARTER. 15-35.
- [4] Safaat, N. (2012). *Pemrograman Aplikasi Mobil Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.
- [5] Pender, Thomas (2002) UML Weekend Crash Course, Wiley Publishing Inc., Indiana