

**SISTEM PENDUKUNG KEPUTUSAN REKOMENDASI BARANG
ONLINE MENGGUNAKAN METODE *WEB SCRAPPING* DAN
*FUZZY-SIMPLE ADDITIVE WEIGHTING (SAW)***

(Studi kasus: *E-commerce* Indonesia)

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Boni Saputra

NIM: 125150100111014



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

PENGESAHAN

SISTEM PENDUKUNG KEPUTUSAN REKOMENDASI BARANG *ONLINE*
MENGUNAKAN METODE *WEB SCRAPING* DAN *FUZZY – SIMPLE ADDITIVE*
WEIGHTING (SAW)
(STUDI KASUS: *E-COMMERCE* INDONESIA)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh:

Boni Saputra

NIM: 125150100111014

Skripsi ini telah disetujui oleh dosen pembimbing pada
28 Juli 2016

Dosen Pembimbing I

Dosen Pembimbing II

Edy Santoso, S.Si., M.Kom
NIP: 19740414 200312 1 004

Laili Muflikhah, S.Kom., M.Sc
NIP: 19741113 200501 2 001

Mengetahui

Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T., M.T., Ph.D
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

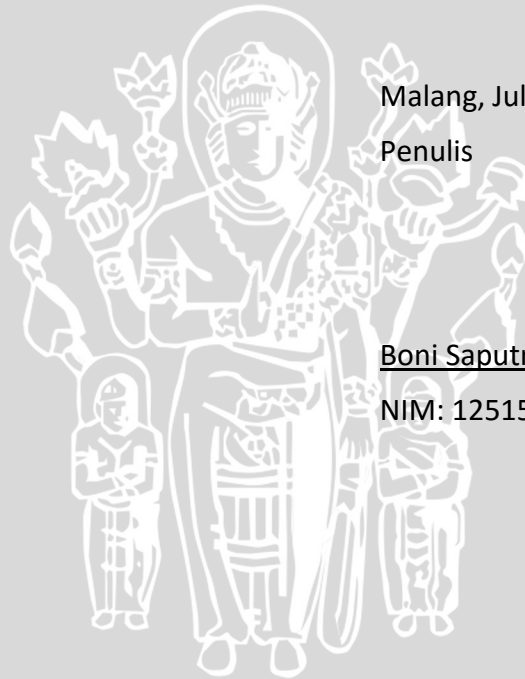
Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, Juli 2016

Penulis

Boni Saputra

NIM: 125150100111014



KATA PENGANTAR

Puja dan puji syukur penulis hantarkan kehadirat Allah SWT, karna berkat rahmat dan hidayat Nya, penulis dapat menyelesaikan penulisan skripsi. Shalawat beserta salam tidak hentinya penulis panjatkan kepada junjungan besar dan kyai umat manusia, Nabi Muhammad SAW, beserta keluarga, dan para sahabat beliau, amin.

Skripsi dengan judul Sistem Pendukung Keputusan Rekomendasi Barang *Online* Menggunakan Metode *Web scraping* dan *Fuzzy-Simple Additive Weighting (SAW)* (Studi kasus: *E-commerce* Indonesia). Penulis susun untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer (S.Kom) pada Jurusan Informatika/Illmu Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya.

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu dalam penyusunan skripsi ini. Secara khusus penulis sampaikan rasa terima kasih kepada:

1. Edy Santoso, S.Si., M.Kom., selaku dosen pembimbing 1 yang telah bersedia untuk membimbing, memberikan arahan dan motivase kepada penulis, sehingga skripsi ini dapat terselesaikan.
2. Lailil Muflikhah, S.Kom., M.Sc., selaku dosen pembimbing 2, yang telah memberikan saran dan kritik yang membangun serta membimbing penulis selama proses penyusunan skripsi ini.
3. Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D., selaku Dekan Fakultas Ilmu komputer, Universitas Brawijaya.
4. Tri Astoto Kurniawan, S.T., M.T., P.D., selaku Ketua Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
5. Agus Wahyu Widodo, S.Kom., M.Cs., selaku Ketua Program Studi Informatika Fakultas Ilmu komputer, Universitas Brawijaya.
6. Seluruh dosen dan karyawan Fakultas Ilmu Komputer Universitas Brawijaya, atas arahan dan bantuannya hingga penulis dapat menyelesaikan skripsi ini.
7. Bapak Efendi dan Ibu Hadriyati, kedua orang tua penulis yang telah mendukung segala kebutuhan, selama proses perkuliahan hingga penulisan skripsi ini selesai.
8. Sonia Hadiva yang selalu menyemangati dan membantu penulis, dalam menyusun penulisan naskah skripsi ini.
9. Teman seperjuangan, Fahmi Arief Hidayat, Kikit Maulana PUA, Rizky Happy Ananda dan semua sahabat yang telah bersedia memberi bantuan kepada penulis.

10. Seluruh rekan-rekan seangkatan Jurusan Informatika/Illmu Komputer Fakultas Ilmu Komputer Universitas Brawijaya.

11. Semua pihak yang tidak dapat penulis sebutkan satu-persatu namanya, yang terlibat secara langsung ataupun tidak langsung dan telah menyumbangkan semangatnya dalam penulisan skripsi ini.

Penulis menyadari bahwa masih banyak kekurangan, dalam penulisan naskah skripsi ini. Oleh karena itu penulis mengharapkan saran dan kritiknya, untuk menutupi kekurangan yang terdapat pada skripsi ini, melalui *email* penulis. Penulis berharap, semoga judul yang dibahas dapat menambah ilmu dan wawasan bagi pembaca terutama dari Fakultas Ilmu Komputer, Universitas Brawijaya dan khususnya penulis sendiri.

Malang, Juli 2016

Penulis



ABSTRAK

Boni Saputra, 2016. Sistem Pendukung Keputusan Rekomendasi Barang Online Menggunakan Metode Web Scrapping dan Fuzzy-Simple Additive Weighting (SAW) (Studi Kasus: E-Commerce Indonesia)

Dosen Pembimbing: Edy Santoso, S.Si, M.Kom dan Lailil Muflikhah, S.Kom, M.Sc

Electronic Commerce (E-commerce) adalah media yang digunakan untuk melakukan transaksi jual beli *online*, *E-commerce* mengalami pertumbuhan yang sangat pesat pada satu dekade terakhir, ditunjukkan dengan maraknya iklan *E-commerce* yang tayang di televisi, tingginya *traffic* kunjungan situs-situs *E-commerce*, dan tingginya jumlah instalasi aplikasi *E-commerce* di perangkat bergerak. Seiring dengan laju pertumbuhan *E-commerce* yang begitu pesat di dunia maupun di Indonesia, memungkinkan terjadinya tindak kejahatan, seperti penipuan dan pemalsuan barang yang akan diperjualbelikan.

Untuk mengatasi tindak kejahatan *E-commerce*, dibutuhkan sistem yang mampu me-rekomendasikan barang terbaik pada setiap *E-commerce*, sistem yang dibangun sebagai solusi untuk meminimalisir tindak kejahatan *E-commerce* adalah sistem Pendukung Keputusan Rekomendasi Barang Online Menggunakan Metode *Web scraping* dan *Fuzzy Simple Additive Weighting*. Metode *Web scraping* digunakan untuk mendapatkan data *E-commerce* dan metode *Fuzzy Simple Additive Weighting* digunakan untuk merekomendasikan barang terbaik.

Dari hasil pengujian dan analisa yang telah dilakukan, fungsional sistem memiliki nilai akurasi 100%. Artinya sistem dapat berjalan dengan baik, sesuai dengan perancangan. Perubahan jumlah data atau sensitivitas berpengaruh pada nilai kesesuaian akurasi rekomendasi, semakin banyak data yang diuji maka semakin tinggi nilai kesesuaian rekomendasi.

Kata kunci: *E-commerce*, jual-beli *online*, *Web scraping*, *Simple Additive Weighting*, *Fuzzy*

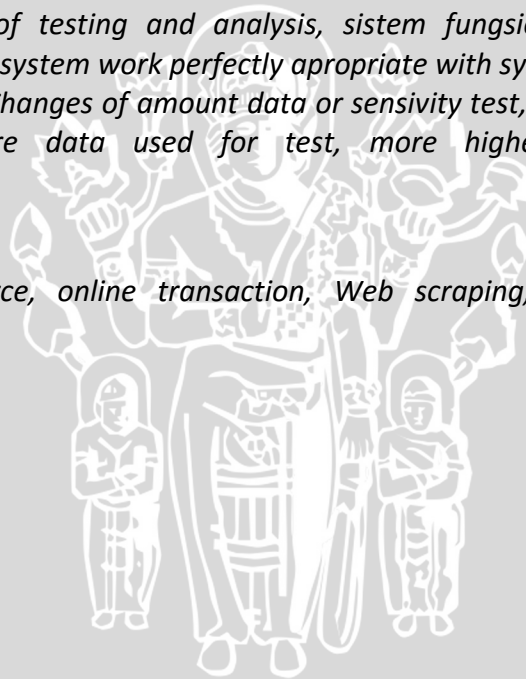
ABSTRACT

Electronic Commerce (E-commerce) is one of medium used to support online transaction, today E-commerce is growing rapidly, it's indicated by the rise of E-commerce advertising through television, increasing of E-commerce website traffic in the internet and mobile apps installation. As far as E-commerce growing up in the world, especially in Indonesia, that is a possibility the criminal able to do crime on E-commerce as much as he want.

To decrease E-commerce crime such a fabrication, needed a system to able recommending best stuff for each E-commerce. The system is Decision Support System for Recommendation Online Stuff using Web scraping Methode and Fuzzy Simple Additive Weighting. Web scraping metode is used to get data from the E-commerce, and Fuzzy Simple Additive Weighting is used to recommend the best stuff.

Based on result of testing and analysis, sistem fungsionalitas has 100% accuracy, it's means system work perfectly appropriate with system design which have been designed. Changes of amount data or sensivity test, it will be affect to the accuracy, more data used for test, more higher accuracy of recommendation.

Keywords: *E-commerce, online transaction, Web scraping, Simple Additive Weighting, Fuzzy*



DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiv
DAFTAR KODE SUMBER.....	xvi
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	3
1.7 Jadwal Pelaksanaan/Penelitian	4
BAB 2 LANDASAN KEPUSTAKAAN	6
2.1 Kajian Pustaka	6
2.2 <i>E-commerce</i>	7
2.2.1 Pengertian <i>E-commerce</i>	7
2.2.2 Sejarah <i>E-commerce</i>	7
2.2.3 <i>E-commerce</i> di Indonesia.....	8
2.3 <i>Web scraping</i>	9
2.4 Sistem Pendukung Keputusan (SPK).....	10
2.4.1 Pengertian Sistem Pendukung Keputusan.....	10
2.4.2 Komponen SPK.....	11
2.4.3 Tahapan Sistem Pendukung Keputusan.....	12
2.5 <i>Fuzzy</i>	12
2.5.1 Pengertian Logika <i>Fuzzy</i>	13

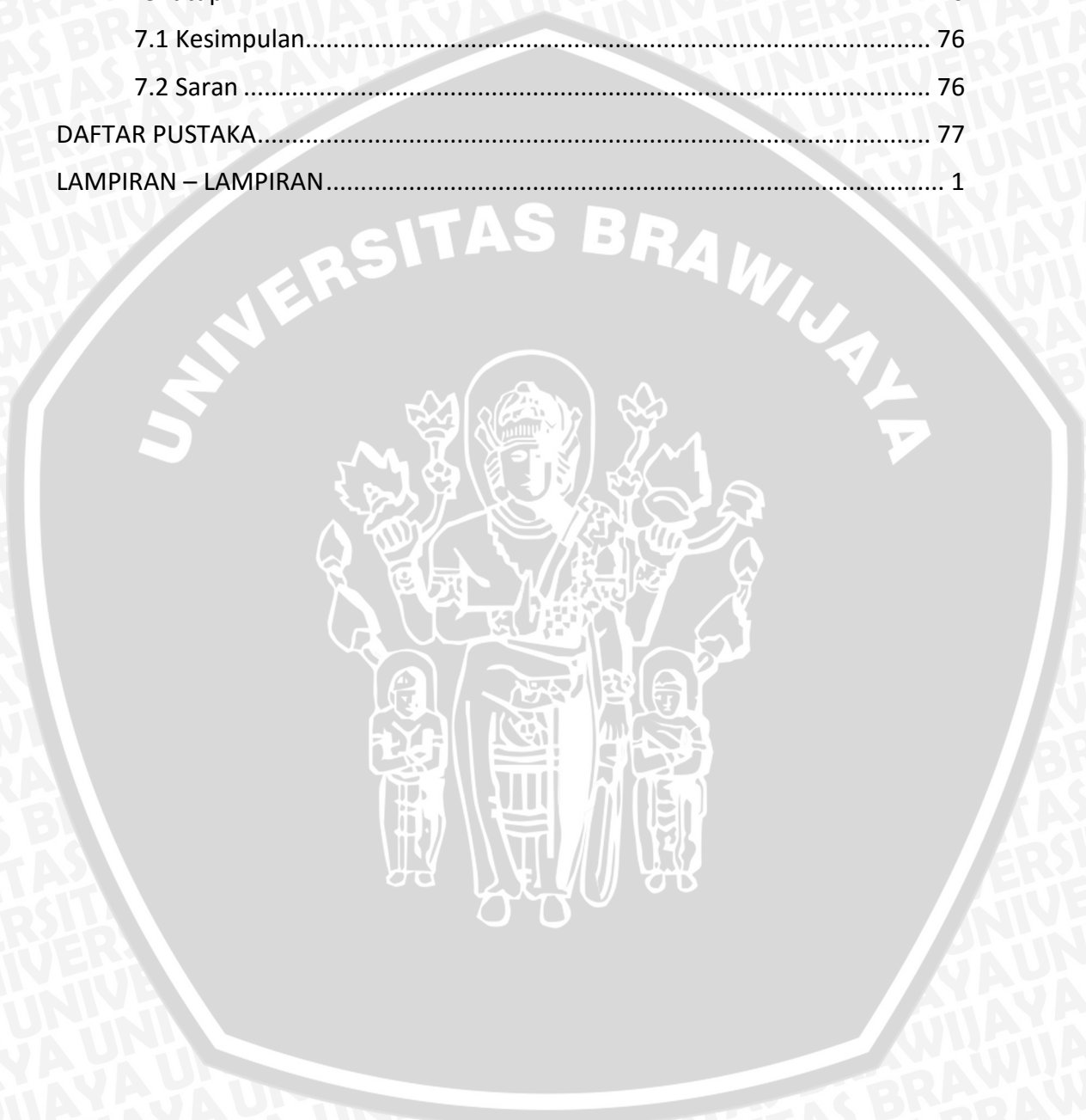
2.5.2 Derajat nilai <i>Fuzzy</i>	13
2.5.3 Kelebihan dan kekurangan.....	14
2.6 <i>Simple Additive Weighting</i> (SAW).....	14
BAB 3 METODOLOGI	16
3.1 Studi Literatur	16
3.2 Pengumpulan Data	16
3.3 Analisis Kebutuhan	16
3.4 Perancangan Sistem.....	17
3.4.1 Model Perancangan Sistem	17
3.5 Implementasi Sistem	18
3.6 Pengujian Sistem.....	19
3.7 Kesimpulan.....	19
BAB 4 PERANCANGAN	20
4.1 Analisis Kebutuhan Perangkat Lunak.....	20
4.1.1 Identifikasi Aktor	20
4.1.2 Daftar Kebutuhan.....	20
4.1.3 Diagram Use Case	21
4.1.4 Diagram Class	22
4.2 Manajemen Data	23
4.2.1 Entity Relationship Diagram.....	23
4.2.2 Tabel Basis Data	24
4.2.3 Basis Pengetahuan	26
4.2.4 Bobot Preferensi dan Kriteria Data	27
4.3 Perancangan Diagram Alir dan Algoritma	30
4.3.1 Algoritma <i>Scraping</i> data barang	31
4.3.2 Algoritma Pencarian Data	33
4.3.3 Algoritma Normalisasi Data	35
4.3.4 Algoritma Preferensi Data.....	38
4.4 Antar Muka Pengguna	42
4.4.1 Halaman Utama Pencarian	42
4.4.2 Halaman Hasil Pencarian Barang	42
4.4.3 Halaman Kategori Barang	44



4.4.4 Halaman Hasil Rekomendasi.....	44
4.4.5 Halaman Detail Barang	46
4.4.6 Halaman Login Admin	47
4.4.7 Halaman Admin Kelola Data Barang	47
4.4.8 Halaman Admin <i>Scraping</i> Barang.....	48
BAB 5 IMPLEMENTASI	49
5.1 Batasan Implementasi	49
5.1.1 Batasan Perangkat Keras (Hardware)	49
5.1.2 Batasan Perangkat Lunak (Software).....	49
5.1.3 Batasan Sumber Daya Manusia (Brainware)	50
5.2 Implementasi <i>Fuzzy Simple Additive Weighting</i>	50
5.2.1 Implementasi Algoritma <i>Fuzzy</i>	50
5.2.2 Implementasi Algoritma Normalisasi.....	54
5.2.3 Implementasi Algoritma Preferensi.....	56
5.3 Implementasi Proses <i>Scraping</i> Data	58
5.3.1 <i>Scraping</i> Situs Bukalapak	59
5.3.2 <i>Scraping</i> Situs Elevenia.....	61
5.4 Implementasi Antarmuka	64
5.4.1 Antarmuka Halaman Utama Pencarian	64
5.4.2 Antarmuka Halaman Barang	64
5.4.3 Antarmuka Halaman Detail Barang.....	65
5.4.4 Antarmuka Halaman Perankingan	65
5.4.5 Antarmuka Halaman Kategori.....	66
5.4.6 Antarmuka Halaman Login <i>Admin</i>	66
5.4.7 Antarmuka Halaman <i>Scraping</i> Data Barang.....	67
5.4.8 Antarmuka Halaman Kelola Barang	68
BAB 6 PENGUJIAN DAN ANALISA	69
6.1 Pengujian Fungsional	70
6.1.1 Skenario Pengujian Fungsional	70
6.1.2 Hasil Pengujian Fungsional.....	70
6.1.3 Analisa Hasil Pengujian Fungsional	73
6.2 Pengujian Sensitivitas	73



6.2.1 Skenario Pengujian Sensitivitas.....	73
6.2.2 Hasil Pengujian Sensivitas	73
6.2.3 Analisa Hasil Pengujian Sensitivitas	75
BAB 7 Penutup	76
7.1 Kesimpulan.....	76
7.2 Saran	76
DAFTAR PUSTAKA.....	77
LAMPIRAN – LAMPIRAN.....	1



DAFTAR TABEL

Tabel 1.1 Rencana Jadwal Penelitian	4
Tabel 2.1 Peringkat Traffic <i>E-commerce</i> Indonesia.....	8
Tabel 4.1 Identifikasi Aktor	20
Tabel 4.2 Kebutuhan Fungsional.....	21
Tabel 4.3 Tabel SK_Barang.....	24
Tabel 4.4 Tabel SK_Kategori.....	25
Tabel 4.5 Tabel SK_Kriteria	25
Tabel 4.6 Kriteria dan penilaian barang.....	26
Tabel 4.7 Parameter kriteria harga	27
Tabel 4.8 Parameter kriteria berat barang	28
Tabel 4.10 Parameter kriteria stok barang	28
Tabel 4.11 Parameter kriteria lokasi pengiriman barang	28
Tabel 4.12 Parameter kriteria deskripsi barang.....	29
Tabel 4.13 Parameter kriteria <i>rating/review</i>	29
Tabel 4.14 Parameter kriteria penjualan unit barang.....	30
Tabel 4.15 Normalisasi data.....	37
Tabel 4.16 Nilai preferensi setiap alternatif	39
Tabel 4.17 Nilai total preferensi alternatif (Vi) sebelum diranking	40
Tabel 4.18 Nilai total preferensi alternative (Vi) setelah diranking.....	41
Tabel 6.1 Rancangan Tabel Pengujian Fungsional	70
Tabel 6.2 Rancangan Tabel Pengujian Sensivitas.....	70
Tabel 6.3 Pengujian Halaman Utama Pencarian.....	70
Tabel 6.4 Pengujian Halaman Rekomendasi.....	71
Tabel 6.5 Pengujian Halaman Detil Barang.....	71
Tabel 6.6 Pengujian Halaman Kategori Barang.....	71
Tabel 6.7 Pengujian Halaman Login	72
Tabel 6.8 Pengujian Halaman Kelola Barang	72
Tabel 6.9 Pengujian Halaman <i>Scarping</i> Data	72
Tabel 6.10 Pengujian Sensivitas Kata Kunci Iphone 5	73
Tabel 6.11 Sensivitas Kata Kunci Samsung.....	74



DAFTAR GAMBAR

Gambar 2.1 Indonesia <i>E-commerce</i> facts	9
Gambar 2.2 Ilustrasi Mekanisme <i>Web scraping</i>	10
Gambar 2.3 Model Konseptual DSS	12
Gambar 2.4 Diagram Logika <i>Fuzzy</i>	13
Gambar 3.1 Model Perancangan Sistem.....	18
Gambar 4.1 <i>Use Case Diagram</i>	22
Gambar 4.2 Diagram Class	23
Gambar 4.3 <i>Entity Relationship Diagram</i>	24
Gambar 4.4 Diagram alir metode <i>Simple Additive Weighting</i>	31
Gambar 4.5 Diagram alir <i>scraping</i> situs <i>E-commerce</i>	32
Gambar 4.6 Algoritma <i>scraping</i> data barang.....	33
Gambar 4.7 Diagram alir pencarian data barang.....	34
Gambar 4.8 Algoritma pencarian data barang	34
Gambar 4.9 Diagram alir normalisasi.....	35
Gambar 4.10 Algoritma Normalisasi.....	36
Gambar 4.11 Diagram alir nilai preferensi.....	38
Gambar 4.12 Algoritma Preferensi	39
Gambar 4.13 Halaman Home.....	42
Gambar 4.14 Halaman Hasil Pencarian dan Kategori	43
Gambar 4.15 Halaman Kategori Barang	44
Gambar 4.16 Halaman Rekomendasi Barang	45
Gambar 4.17 Halaman Detail Barang.....	46
Gambar 4.18 Halaman Login Admin	47
Gambar 4.19 Halaman Kelola Barang	48
Gambar 4.20 Halaman <i>Scraping</i> Barang	48
Gambar 5.1 Antarmuka Halaman Utama Pencarian.....	64
Gambar 5.2 Antarmuka Halaman Barang	65
Gambar 5.3 Antarmuka Halaman Detil Barang.....	65
Gambar 5.4 Antarmuka Halaman Perankingan	66
Gambar 5.5 Antarmuka Halaman Kategori.....	66

Gambar 5.6 Antarmuka Halaman Login..... 67

Gambar 5.7 Antarmuka Halaman Scraping Data Barang..... 67

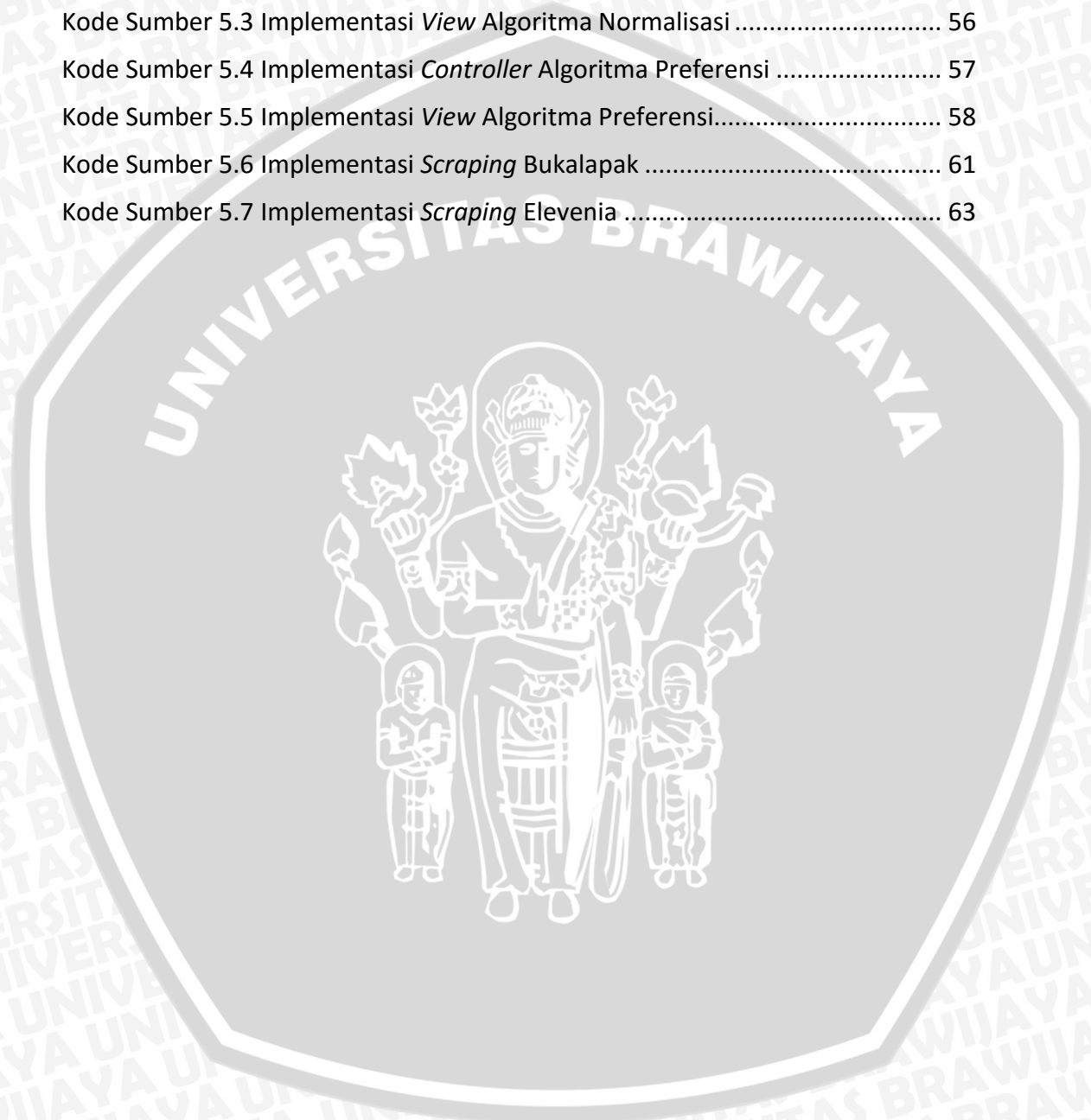
Gambar 5.8 Antarmuka Halaman Kelola Barang..... 68

Gambar 6.1 Diagram Alir Pengujian..... 69



DAFTAR KODE SUMBER

Kode Sumber 5.1 Implementasi <i>Fuzzy</i>	53
Kode Sumber 5.2 Implementasi <i>Controller</i> Algoritma Normalisasi	54
Kode Sumber 5.3 Implementasi <i>View</i> Algoritma Normalisasi	56
Kode Sumber 5.4 Implementasi <i>Controller</i> Algoritma Preferensi	57
Kode Sumber 5.5 Implementasi <i>View</i> Algoritma Preferensi.....	58
Kode Sumber 5.6 Implementasi <i>Scraping</i> Bukalapak	61
Kode Sumber 5.7 Implementasi <i>Scraping</i> Elevenia	63



BAB 1 PENDAHULUAN

1.1 Latar belakang

Dewasa ini *Electronic Commerce (E-commerce)* atau perniagaan *online*, mengalami perkembangan yang pesat. Hal ini disebabkan oleh penggunaan internet yang meningkat tajam setiap tahunnya, dan inovasi teknologi yang terus berkembang. *E-commerce* merupakan alternative pilihan, untuk mengatasi permasalahan ekonomi, yaitu dengan membantu tumbuh pesatnya usaha kecil menengah (UKM). Pada awal perkembangannya *E-commerce* di Indonesia, dipandang sebelah mata, dikarenakan tidak adanya pihak perantara dalam melakukan transaksi keuangan yang aman, sehingga banyak terjadi penipuan, barang palsu, dan tidak sesuai pesanan. Hingga dewasa ini, dikenal sistem pembayaran elektronik baru, yaitu sistem *escrow* atau lebih umum dikenal sebagai rekening bersama.

Sistem *escrow* mulai ramai dipergunakan pada salah satu forum terbesar di Indonesia, Kaskus. Saat melakukan transaksi melalui Kaskus banyak hal yang harus dipertimbangkan, diantaranya jumlah postingan, berapa “cendol” (semacam *badges positive*), apakah penjual termasuk *recommended seller*, atau penjual masuk kedalam list penjual bermasalah, dan aspek pendukung lainnya. Sejatinnya forum seperti Kaskus tidak diperuntukkan untuk transaksi jual-beli melainkan untuk berbagi informasi dan media komunikasi (Kaskus, 2016). Sehingga untuk mengatasi masalah tersebut diperlukan model *E-commerce* dengan sistem *escrow* untuk menampung penjual *online*, dan ribuan produk didalamnya, dikenal dengan sebutan *marketplace*. *E-commerce* dengan model *marketplace* di Indonesia diantaranya Bukalapak dan Elevenia.

Setelah dikenalnya sistem *escrow*, dan munculnya situs *E-commerce* dengan model *marketplace* sebagai wadah untuk melakukan transaksi elektronik, masalah barupun muncul, seperti jumlah barang yang diposting hingga mencapai jutaan barang pada situs *E-commerce*. Hal ini tidak menutup kemungkinan terjadinya kasus pemalsuan barang yang dilakukan oleh oknum yang tidak bertanggung jawab. Salah satu kasus pemalsuan yang terjadi pada situs *E-commerce* terbesar di Indonesia adalah kasus pembelian iPhone 6 Plus, pada saat barang diterima, konsumen mendapatkan sabun batang Nuvo bukan iPhone 6 Plus seperti yang diharapkan. Kasus lainnya adalah transaksi *Cash on Delivery* melalui situs jejaring iklan OLX, pembeli dirampok saat transaksi pembelian 1 unit mobil berlangsung. Tidak sedikit cerita yang berakhir duka mulai dari gagal bertransaksi hingga nyawa melayang (Kompas, 2016).

Rekomendasi barang sangat dibutuhkan untuk mengurangi angka penipuan dan perampokan khususnya dalam transaksi *online*. Dalam dunia digital, sistem rekomendasi memiliki peran penting untuk menentukan sebuah pilihan. Salah satu metode yang digunakan untuk rekomendasi adalah *Fuzzy Simple additive*

weighting (F-SAW). Penelitian ini menggunakan metode *Fuzzy Simple additive weighting* dikarenakan metode ini efektif merekomendasikan suatu permasalahan yang cukup kompleks, data dalam jumlah banyak, dan mampu mengkalkulasi banyak kriteria pertimbangan (Usito, N.J, 2013).

Untuk mendapatkan data dari internet dan mengkalkulasikannya kedalam metode F-SAW, dibutuhkan sistem yang mampu mengambil data dari internet. *Web scraping* merupakan salah satu proses yang digunakan untuk mengambil data dari Internet. Penggunaan *web scraping* sangat dibutuhkan oleh penyedia layanan jasa berbasis Internet, mulai dari analisis, *indexing*, perangkian situs, dan berbagai manfaat lainnya, oleh karna itu *web scraping* digunakan untuk mendapatkan data barang pada situs *E-commerce*.

Dari permasalahan diatas, dibutuhkan sebuah sistem yang mampu memilih barang dengan *rating* dan *review* yang baik, lokasi pengiriman yang dekat sehingga menghemat biaya pengiriman, jumlah penjualan, dan beberapa aspek penting lainnya. Sistem yang dimaksud adalah sistem pendukung keputusan rekomendasi barang *online* pada situs *E-commerce* dengan model bisnis marketplace seperti yang telah diutarakan diatas. Dimana sistem ini nantinya akan menggunakan metode *Web Scrapping* untuk mendapatkan data dan *Fuzzy-Simple additive weighting* (F-SAW) untuk proses pengambilan keputusannya. *Fuzzy-Simple additive weighting* (F-SAW) dipilih karena efektivitasnya dan tingkat akurasi yang tinggi, dan kemudahan dalam penulisan kode sumber.

1.2 Rumusan masalah

Berdasarkan latar belakang yang dipaparkan di atas, maka permasalahan yang akan dibahas diantaranya:

1. Apakah metode *web scraping* mampu mendapatkan data barang *online* dan metode *fuzzy-simple additive weighting* dapat merekomendasikan barang?
2. Berapa persen nilai akurasi pengujian fungsional sistem?
3. Bagaimana hubungan jumlah data dan tingkat akurasi kesesuaian rekomendasi barang?
4. Apa pengaruh kata kunci terhadap uji sensitivitas pada sistem rekomendasi barang *online*?

1.3 Tujuan

Tujuan umum:

Merancang dan mengimplementasikan sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scrapping* dan *fuzzy-simple additive weighting*.

Tujuan khusus:

1. Merancang sistem rekomendasi barang *online*, dengan metode *web scrapping* dan *fuzzy-simple additive weighting*.

2. Mendapatkan data barang *online* dari situs *E-commerce* Indonesia.
3. Memodelkan data barang kedalam logika *Fuzzy*
4. Merekomendasikan barang terbaik dengan tingkat akurasi yang tinggi

1.4 Manfaat

Penulisan skripsi ini diharapkan mampu mengatasi kekecewaan pengguna atau yang terlibat langsung dalam transaksi *online* terhadap barang yang akan dibeli, dengan cara merekomendasikan barang yang berdasarkan kriteria barang, yang sering menjadi pertimbangan pengguna layanan *E-commerce* sebelum melakukan transaksi.

1.5 Batasan masalah

Untuk menghindari topik bahasan, keluar dari topik bahasan yang telah dirumuskan sebelumnya maka diperlukan batasan-batasan. Batasan masalah dalam penelitian ini adalah:

1. Data objek penelitian hanya dibatasi pada 2 situs besar *E-commerce* di Indonesia, yaitu Bukalapak, Elevenia.
2. Parameter yang digunakan untuk mengambil keputusan barang yang tepat dibatasi pada 7 kriteria, yaitu harga barang, berat barang, lokasi pengiriman, deskripsi barang, *rating/review*, stok barang, penjualan unit barang.
3. Sistem pendukung keputusan menggunakan metode *Fuzzi - Simple Additive Wiegthing*, penentuan nilai bobot preferensi ditentukan oleh pengguna, bobot sub-kriteria menggunakan metode *Range Order Centroid*.
4. Dirancang dan diimplementasikan dalam bentuk halaman *web*, menggunakan Bahasa pemograman PHP, dan MySQL untuk penyimpanan *database*.

1.6 Sistematika pembahasan

Untuk mencapai tujuan yang diharapkan, maka sistematika laporan yang disusun dalam tugas akhir ini adalah sebagai berikut:

BAB I : Pendahuluan

Pada Bab I membahas tentang latar belakang masalah penelitian, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika penulisan skripsi ini.

BAB II : Landasan Kepustakaan

Bab Landasan Kepustakaan berisi tentang definisi *E-commerce* dan menjelaskan tentang teori-teori beserta pemodelan metode yang digunakan dalam penelitian ini. Serta membahas penelitian-penelitian sebelumnya dengan metode yang sama yang didapat dari berbagai macam referensi untuk menunjang penelitian dalam penulisan skripsi ini.

BAB III : Metodologi

Bab ini berisi metode yang digunakan dalam penelitian skripsi yang terdiri dari studi literatur, pengumpulan data, perancangan sistem, implementasi sistem, pengujian sistem, dan kesimpulan.

BAB IV : Analisa dan Perancangan Sistem

Pada bab analisa dan perancangan sistem membahas tentang Analisis kebutuhan dan perancangan aplikasi yang dibuat meliputi deskripsi aplikasi, spesifikasi kebutuhan dan perancangan sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scraping* dan *Fuzzy-Simple Additive Weighting (SAW)*.

BAB V : Implementasi

Bab V membahas tentang lingkungan implementasi misalnya bahasa pemrograman yang dipakai dan spesifikasi sistem dari perangkat keras yang dipakai. Serta algoritma yang diimplementasikan pada penelitian ini juga akan dijabarkan.

BAB VI : Pengujian dan Analisis

Bab pengujian dan analisis ini menjelaskan pengujian akurasi pada sistem pendukung keputusan, dan analisis terhadap sistem pendukung keputusan yang telah direalisasikan dan telah memenuhi kriteria.

BAB VII : Penutup

Pada Penutup, memuat tentang kesimpulan yang diperoleh dari pembuatan dan pengujian sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scraping* dan *Fuzzy-Simple Additive Weighting (F-SAW)* yang dikembangkan pada tugas ini, dan kelebihan kekurangan serta saran-saran untuk pengembangan lebih lanjut.

1.7 Jadwal Pelaksanaan/Penelitian

Kegiatan ini akan dilaksanakan selama empat bulan dengan jadwal kerja seperti ditunjukkan dalam tabel berikut:

Tabel 1.1 Rencana Jadwal Penelitian

No	Kegiatan				
		I	II	III	IV
1	Identifikasi kebutuhan sistem	■			
2	Perancangan sistem		■		

3	Implementasi				
4	Pengujian sistem				
5	Penulisan Skripsi				



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Beberapa penelitian lain yang telah dilakukan dengan menggunakan metode *Simple additive weighting* (SAW) antara lain dilakukan oleh Nugroho Joko Usito, yang membuat sistem pendukung keputusan penilaian proses belajar mengajar menggunakan metode *simple additive weighting* (SAW). *Inputan* kriteria dari penelitian ini adalah tingkat kehadiran mengajar, ketepatan mulai dan mengakhiri kuliah, kesesuaian materi dengan silabus, kemudahan penyampaian materi untuk dipahami, motivasi belajar, penggunaan alat bantu untuk memperjelas materi, melayani, memberi perhatian dalam komunikasi dua arah, membantu, akomodatif pengajar, dan pengajar mudah untuk ditemui, serta memiliki pengetahuan aktual dalam pembelajaran. Hasil penelitian menunjukkan bahwa sistem mampu menampilkan data mata kuliah dan kelas yang diajar, berikut hasil PBM dosennya (Usito, N.J, 2013).

Kemudian, Kaho, M.L.R, Tanaamah,A.R, dan Wowor A.D, melakukan penelitian dengan judul penerapan metode *Simple Additive Weighting* pada penentuan tingkat kesejahteraan penduduk propinsi Nusa Tenggara Timur. Kriteria inputan dari sistem ini adalah kepadatan penduduk, persentase penduduk, persentase angka kesakitan penduduk, persentase pengeluaran rumah tangga yang menggunakan listrik, persentase perkapita, dan persentase kemiskinan. Setelah menetapkan bobot dan kriteria, peneliti menggunakan *fuzzy* untuk memodelkan data, sistem dapat dikatakan berhasil karena hasil dari data manual dan hasil yang diberikan sistem sama. (Kaho, M.L.R, Tanaamah,A.R, dan Wowor A.D, 2013).

Selanjutnya Rinaldi G.E, melakukan penelitian dengan judul penerapan metode *Simple Additive Weighting* (SAW) untuk sistem pendukung keputusan penentuan penerimaan beasiswa bantuan siswa miskin (BSM) pada SMA Negeri 1 Subah Kab.Batang. Inputan dari penelitian ini adalah gaji orang tua, jumlah tanggungan, nilai rapor, kepribadian, dan prestasi. Hasil yang didapatkan dari penelitian ini adalah sistem yang dibangun dapat membantu mempercepat proses penyeleksian beasiswa. Sistem yang dibangun dapat mengurangi kesalahan dalam menentukan penerima calon beasiswa. (Rinaldi G.E, 2013).

Berikutnya, penelitian yang dilakukan oleh Yohana D.L.W., Maya S.R, Heni R, dengan judul penelitian Sistem Pendukung Keputusan Penentuan Karyawan Terbaik Menggunakan Metode *Simple Additive Weighting* (SAW) Studi Kasus PT. Pertamina Ru li Dumai. Peneliti menggunakan metode SAW mulai dari pembobotan hingga perankingan tanpa ada bantuan metode lain, yang menjadi kriteria inputan pada penelitian ini adalah prestasi, aktif, *safety*, sehat, walaupun kriteria terbilang sedikit hasil yang didapatkan ialah sistem berhasil menentukan karyawan terbaik (Lulu, Y.D, Sari, R.M, Rachmawati, H. 2013).

Berikutnya penelitian yang dilakukan oleh Boettcher I., dengan judul, *Automatic data collection on the internet (web scraping)*. Penulis menganalisis penggunaan *web scraping* untuk mengambil data harga, yang dilakukan oleh

badan statistika Austria. Hasil yang didapatkan teknologi *web scraping* dapat digunakan untuk meningkatkan kualitas data statistik tetapi juga harus mempertimbangkan legalitas data (Boettcher I., 2015).

Berikutnya penelitian yang dilakukan oleh Fernandez J.I, Blasco J., Iglesias C.A., Garijo M., dengan judul, *A semantic scraping for web resources Applying Linked Data to Web Page Screen Scraping*. Penulis menggunakan *framework Semantic Scraping Framework* untuk mengolah data tampilan laman *web*. Penulis menjelaskan bahwa *web scraping* memiliki tingkatan, *Scraping Service Level*, *Semantic Service Level*, dan *Syntatic Scraping Level*. Kesimpulan penggunaan *framework* efektif untuk mengatasi masalah *web scraping* dibandingkan dengan tanpa menggunakan *framework* dan atau *library*. (Fernandez J.I, Blasco J., Iglesias C.A., Garijo M., 2009).

2.2 E-commerce

Sub bab 2.2 ini akan menjelaskan tentang pengertian *E-commerce*, sejarah *E-commerce*, dan *E-commerce* di Indonesia.

2.2.1 Pengertian E-commerce

E-commerce adalah singkatan dari *Electronic Commerce* yang berarti segala jenis transaksi dilakukan melalui suatu jaringan, jaringan yang dimaksud adalah internet. Dalam satu jaringan internet, penjual dan pembeli terhubung untuk melakukan transaksi jual-beli.

E-commerce memiliki arti yang berbeda bagi orang yang berbeda. Oleh karena itu penulis menyampaikan beberapa sudut pandang dari para ahli mengenai definisi dari *E-commerce*:

- Definisi *E-commerce* oleh Laudon & Laudon (1998), *E-commerce* adalah suatu proses membeli dan menjual produk-produk secara elektronik oleh konsumen dan dari perusahaan ke perusahaan dengan komputer sebagai perantara transaksi bisnis.
- Definisi *E-commerce* menurut David Baum (1999) yang diterjemahkan oleh Onno. W. Purbo: *E-commerce* merupakan satu set dinamis teknologi, aplikasi, dan proses bisnis yang menghubungkan perusahaan, konsumen, dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan, dan informasi yang dilakukan secara elektronik.

2.2.2 Sejarah E-commerce

E-commerce mulai dikembangkan sejak protocol *Hypertext Transfer Protocol Secure* (HTTPS) ditemukan, protocol HTTPS dinyatakan aman untuk melakukan transaksi elektronik dan merahasiakan data penting melalui media *website* atau *World Wide Web* tanpa ada intervensi pihak ketiga. Pada masa awal, *E-commerce* hanyalah seputar penayangan iklan baris pada suatu halaman web, dengan tujuan untuk mempromosikan barang /jasa, seiring berjalannya waktu *E-commerce* pun

berkembang menjadi model retail, yaitu semua transaksi hanya dilakukan oleh situs retail yang bersangkutan dan kemudian model *marketplace* yaitu transaksi dilakukan oleh penjual, situs penyedia dan pembeli. (iese.id, 2016, <http://iese.id/menilik-dua-dekade-lalu-E-commerce-indonesia>, diakses pada 10 Mei 2016)

2.2.3 E-commerce di Indonesia

Perkembangan *E-commerce* di Indonesia mulai berkembang sejak tahun 1996, dengan berdirinya Dyviacom Intrabumi (D-Net) sebagai perintis transaksi *online*. Layanan transaksi berupa *mall online* yang disebut D-Mall, telah menampung sekitar 33 toko *online/merchant*. Produk yang dijual bermacam-macam, mulai dari makanan, aksesoris, pakaian, produk perkantoran, *furniture* dan masih banyak barang yang ditawarkan. Selain itu, berdiri pula *E-commerce-indonesia.com*, tempat penjualan *online* berbasis internet yang memiliki fasilitas lengkap seperti adanya bagian depan toko (*storefront*) dan keranjang belanja (*shopping cart*). (iese.id, 2016, <http://iese.id/menilik-dua-dekade-lalu-E-commerce-indonesia>, diakses pada 11 Mei 2016)

E-commerce di Indonesia semakin berkembang satu dekade terakhir ini, ditandai dengan tingginya peringkat *traffic daily unique visitor* yang dilakukan alexa.com salah satu situs yang dimiliki oleh Amazon *company*. Contoh saja Bukalapak yang menduduki peringkat 10 besar situs paling sering dikunjungi di Indonesia, berikut Tabel 2.2 yang menampilkan peringkat situs *E-commerce* model bisnis *marketplace* di Indonesia:

Tabel 2.1 Peringkat Traffic E-commerce Indonesia

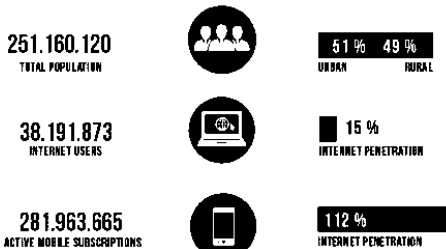
No	Nama E-commerce	Traffic Alexa Rank dunia	Traffic Alexa Rank Indonesia
1.	Bukalapak.com	439	10
2.	Tokopedia.com	542	13
3.	Elevenia.co.id	2164	33

Sumber: [<http://www.alex.com>]

Fakta lain yang mengejutkan adalah jumlah transaksi *E-commerce* pada bulan February 2014 jumlah transaksinya mencapai 36.861.353 transaksi, sedangkan jumlah uang elektronik yang beredar dalam transaksi *E-commerce* pada bulan Juli 2014 mencapai 361.063.000.000 (tiga ratus enam puluh satu triliun enam puluh tiga juta). Berikut data dan grafik mengenai fakta *E-commerce*, disajikan pada Gambar 2.1.

E-COMMERCE FACTS

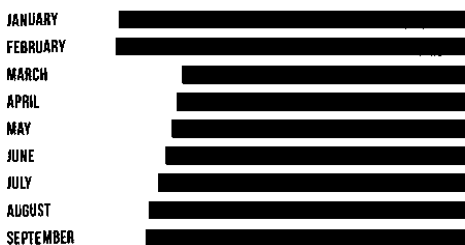
INDONESIA : DATA SNAPSHOT 2014



JUMLAH UANG ELEKTRONIK BEREDAR



TABEL TRANSAKSI UANG ELEKTRONIK



TAHUN 2014, E-COMMERCE AKAN MENJADI SALAH SATU BISNIS DIGITAL YANG AKAN MENJADI TREN. DI INDONESIA SENDIRI, PENETRASI INTERNET YANG TERUS BERTAMBAH MEMBUAT PERTUMBUHAN TOKO ONLINE KIAN SUBUR. DITAMBAH LAGI, VELA ASIA MEMPROYEKSIKAN PASAR E-COMMERCE AKAN MENCAPAI US\$ 18 MILIAR DI TAHUN DEPAN DENGAN PEMBELI ONLINE SEKIRA 37 JUTA ORANG. DI TAHUN INI, DIPERKIRAKAN PASAR E-COMMERCE MENCAPAI US\$ 12 JUTA DENGAN 27 JUTA ORANG PEMBELI ONLINE.

Gambar 2.1 Indonesia E-commerce facts

Sumber: [<http://inovasiipintar.com>]

Kehadiran *E-commerce* sebagai media transaksi baru tentunya menguntungkan banyak pihak, baik pihak konsumen, maupun pihak produsen dan penjual (*retailer*). Dengan menggunakan internet, proses perniagaan dapat dilakukan dengan menghemat biaya dan waktu.

2.3 Web scraping

Web scraping adalah cara yang dilakukan untuk mendapatkan sebagian atau seluruh data dari sebuah halaman *world wide web* pada protokol *Hypertext Transfer Protocol* (HTTP). *Web Scraping* juga dikenal dengan istilah *web harvesting*. Cara ini sangat bermanfaat bagi siapa saja yang membutuhkan data *internet* dalam jumlah banyak, *web scraping* jauh lebih efektif jika dibandingkan memasukkan data dengan cara mengetik, atau *copy-paste*.

Web scraping dapat digunakan untuk berbagai macam keperluan, mulai dari kebutuhan pribadi hingga kebutuhan perusahaan, beberapa manfaat penggunaan *web scraping* adalah:

- *Market research*, *web scraping* dapat membantu perusahaan untuk mendapatkan data yang bersifat *analytic*, untuk kebutuhan analisa pasar.

- Mendapatkan kontak info, web scraping digunakan untuk mendapat informasi kontak pelanggan, misal *e-mail* dan nomor telepon untuk kebutuhan perusahaan.
- Mendapatkan harga dari berbagai situs *E-commerce*, analisa harga barang menjadi pertimbangan bagi kebutuhan perusahaan, dengan mendapatkan informasi harga barang, perusahaan dapat menetapkan harga pada produk serupa.
- Digunakan oleh mesin telusur untuk melakukan pengindeksan halaman *website* dan menelusuri kata kunci tertentu, contoh Google, Bing, Yahoo.
- *Web scraping* juga diterapkan pada hampir semua *browser* untuk menginspeksi elemen HTML pada suatu halaman *website* dengan model *Data Objek Model Tree* (DOM-Tree).

Web scraping dapat dilakukan dengan berbagai macam cara, beberapa teknik *web scraping* yang dikenal ialah *text-grepping* atau *text-mining*, *HTML parser*, *HTML Data Objek Model* atau dengan menggunakan software khusus untuk melakukan *scraping*. Ilustrasi proses pengambilan data, dengan metode *web scraping* dapat dilihat pada Gambar 2.2.



Gambar 2.2 Ilustrasi Mekanisme *Web scraping*

Sumber: [<http://webdata-scraping.com>]

Pada penelitian kali ini, penulis akan menggunakan teknik *HTML parser* dengan bantuan *library simple HTML DOM* untuk mendapatkan data dari situs *E-commerce*.

2.4 Sistem Pendukung Keputusan (SPK)

Sub bab 2.4 ini akan menjelaskan tentang pengertian sistem pendukung keputusan, sejarah, karakteristik dan kemampuan *DSS*, komponen apa saja yang ada di dalamnya, bagaimana tahap-tahapnya, dan kelebihan serta kekurangan dari sistem pendukung keputusan.

2.4.1 Pengertian Sistem Pendukung Keputusan

Sistem pendukung keputusan adalah sistem komputer yang mengolah data menjadi informasi untuk mengambil keputusan dari masalah semi-terstruktur yang spesifik (Ismail, 2013).

Sistem pendukung keputusan memiliki arti yang berbeda bagi orang yang berbeda. Oleh karena itu disini penulis menyampaikan beberapa sudut pandang dari para ahli mengenai Sistem pendukung keputusan:

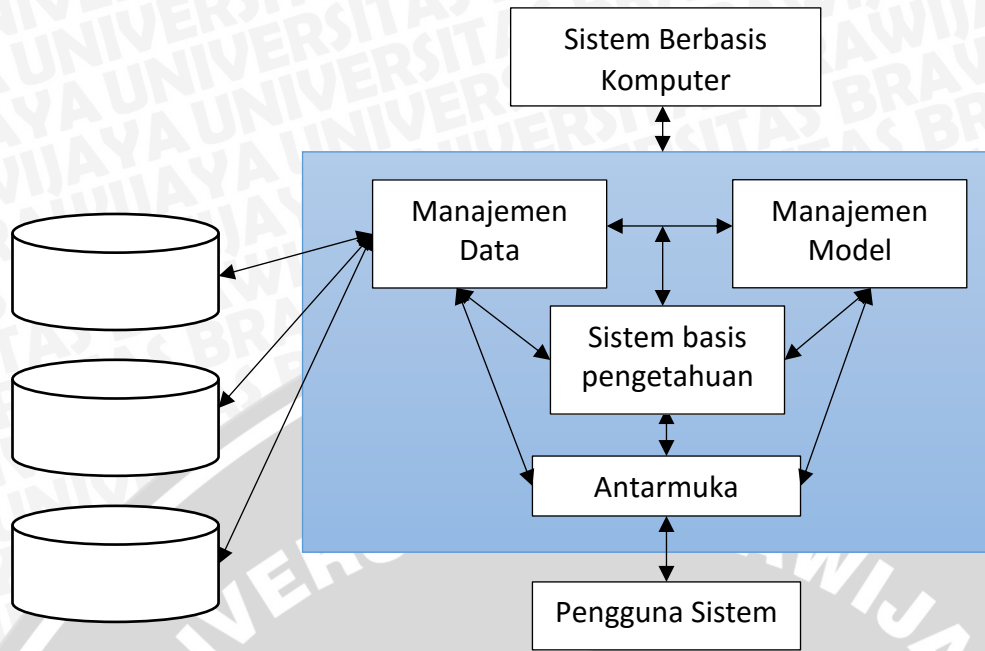
- Menurut Mann dan Watson, Sistem Pendukung Keputusan adalah Sistem yang interaktif, membantu pengambilan keputusan melalui penggunaan data dan model-model keputusan untuk memecahkan masalah-masalah yang sifatnya semi terstruktur dan tidak terstruktur.
- Menurut Maryam Alavi dan H. Albert Napier, Sistem Pendukung Keputusan adalah suatu kumpulan prosedur pemrosesan data dan informasi yang berorientasi pada penggunaan model untuk menghasilkan berbagai jawaban yang dapat membantu manajemen dalam pengambilan keputusan.
- Menurut Litle, Sistem Pendukung Keputusan adalah suatu sistem informasi berbasis komputer yang menghasilkan berbagai alternatif keputusan untuk membantu manajemen dalam menangani berbagai permasalahan yang terstruktur ataupun tidak terstruktur dengan menggunakan data dan model.

2.4.2 Komponen SPK

Turban (2007:109) menyebutkan bahwa DSS mempunyai beberapa komponen untuk membangun sistemnya, antara lain:

1. *Data Management*, berisi data-data relevan termasuk juga dalam *database* yang dikelola oleh software DBMS (*Database Management System*).
2. *Model Management*, model yang terlibat adalah model-model bersifat kuantitatif seperti finansial, statistical, dan berbagai model lainnya sehingga sistem dapat mempunyai kemampuan analitis.
3. *Communication*, tersedia antarmuka antara sistem dengan user sehingga dapat terjalin komunikasi yaitu user mampu memberi perintah pada DSS.
4. *Knowledge Management*, merupakan subsistem optional sehingga bisa menjadi komponen yang berdiri sendiri.

Model konseptual DSS yang dibentuk berdasarkan komponen-komponen yang ada yang kemudian digambarkan seperti pada Gambar 2.3.



Gambar 2.3 Model Konseptual DSS

Sumber: [Prasetyaningrum I., 2014]

2.4.3 Tahapan Sistem Pendukung Keputusan

Menurut Herbert A Simon tahapan pendukung keputusan adalah:

1. Tahap pemahaman (*intelligence phase*). Merupakan proses penelusuran dan pendeteksian dari lingkup problematika serta proses pengenalan masalah. Data masukan diperoleh, diproses dan diuji dalam rangka mengidentifikasi masalah.
2. Tahap perancangan (*Design Phase*). Proses pengembangan dan pencarian alternative tindakan atau solusi yang dapat diambil. Tersebut merupakan representasi kejadian nyata yang mengetahui keakuratan model dalam meneliti masalah yang ada.
3. Tahap pemilihan (*Choice Phase*) pemilihan terhadap diantara berbagai solusi alternatif yang dimunculkan pada tahap perencanaan agar ditentukan atau dengan memperhatikan kriteria-kriteria berdasarkan tujuan yang akan di capai. Tahap Implementasi (*implementation Phase*) penerapan terhadap rancangan sistem yang telah dibuat pada tahap perancangan serta pelaksanaan alternatif tindakan yang telah dipilih pada tahap pemilihan.

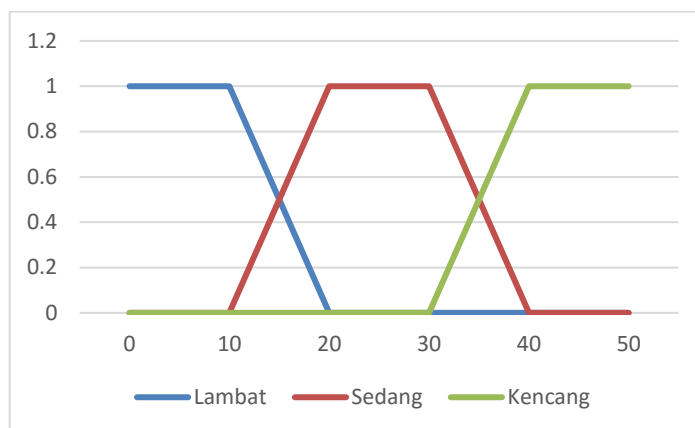
2.5 Fuzzy

Sub Bab 2.5 ini akan menjelaskan tentang pengertian logika *fuzzy*, dan kelebihan serta kekurangan dari logika *fuzzy*.

2.5.1 Pengertian Logika Fuzzy

Logika *fuzzy* diperkenalkan oleh Dr. Lotfi Zadeh dari Universitas California, Berkeley pada 1965. *Fuzzy* secara bahasa dapat diartikan samar, dengan kata lain logika *fuzzy* adalah logika yang samar. Dimana pada logika *fuzzy* suatu nilai dapat bernilai 'true' dan 'false' secara bersamaan. Tingkat 'true' atau 'false' nilai dalam logika *fuzzy* tergantung pada bobot keanggotaan yang dimilikinya. Logika *fuzzy* memiliki derajat keanggotaan rentang antara 0 hingga 1, berbeda dengan logika digital yang hanya memiliki dua keanggotaan 0 atau 1 saja pada satu waktu. Logika *fuzzy* sering digunakan untuk mengekspresikan suatu nilai yang diterjemahkan dalam bahasa (linguistic), semisal untuk mengekspresikan suhu dalam ruangan apakah ruangan tersebut dingin, hangat, atau panas (Earl Cox,1994)

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang input dalam suatu ruang output dan memiliki nilai yang berlanjut. Kelebihan logika *fuzzy* ada pada kemampuan penalaran secara bahasa. Sehingga, dalam perancangannya tidak memerlukan persamaan matematis yang kompleks dari objek yang akan dikendalikan. Contoh pemetaan berdasarkan logika *fuzzy* ialah mengekspresikan kecepatan kedalam tiga kriteria yaitu lambat, sedang, dan kencang. Untuk lebih jelasnya lihat Gambar 2.4.



Gambar 2.4 Diagram Logika Fuzzy

2.5.2 Derajat nilai Fuzzy

Penentuan derajat nilai *fuzzy* pada simple additive weighting dihitung dengan derajat nilai keanggotaan yang dihitung secara manual, atau dengan *range order centroid*, yaitu menentukan nilai derajat keanggotaan melalui perhitungan. Adapun rumus perhitungan range order centroid dapat dilihat pada persamaan (2-1).

$$W_k = \frac{1}{k} \sum_{i=1}^k \left(\frac{1}{i} \right) \quad (2-1)$$

2.5.3 Kelebihan dan kekurangan

Kelebihan Logika *Fuzzy*

- Konsep logika *fuzzy* mudah dimengerti. Konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
- Logika *Fuzzy* sangat fleksibel.
- Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.
- Logika *Fuzzy* mampu memodelkan fungsi-fungsi non linear yang sangat kompleks.
- Logika *fuzzy* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
- Logika *Fuzzy* dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
- Logika *fuzzy* didasarkan pada bahasa alami.

Kekurangan Logika *Fuzzy*

Selain kelebihan yang telah dijelaskan di atas, ternyata *Fuzzy Logic* juga memiliki kekurangan. Dalam mendesain *fuzzy logic*, sering ditemukan kesulitan dalam menentukan preferensi atau parameter agar output yang dihasilkan akurat, yaitu:

- Penentuan model inference harus tepat, Mamdani biasanya cocok untuk masalah intuitive sedangkan sugeno untuk permasalahan yang menangani control
- Kita harus merubah nilai crisp menjadi nilai linguistik. Jumlah dari nilai linguistik yang digunakan harus sesuai dengan permasalahan yang akan kita selesaikan.
- Batas-batas nilai linguistik akan sangat berpengaruh pada akurasi *fuzzy logic*.

2.6 Simple Additive Weighting (SAW)

Metode *Simple Additive Weighting (SAW)* sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari *rating* kinerja pada setiap alternatif pada semua atribut/kriteria (Kusumadewi, 2006). Langkah-langkah yang digunakan untuk pengambilan keputusan dengan metode SAW adalah:

1. Menentukan variabel alternative yaitu A_i
2. Menentukan kriteria yang menjadi acuan dalam pengambilan keputusan yaitu C_j
3. Memberikan nilai *rating* kecocokan setiap alternative pada setiap kriteria

- Menentukan bobot preferensi atau tingkat kepentingan (W) untuk setiap kriteria

$$W = [W_1, W_2, W_3, W_4 \dots W_n] \quad (2-2)$$

- Membuat tabel *rating* kecocokan dari setiap alternative pada setiap kriteria
- Membuat matrik keputusan (X) yang dibentuk dari tabel *rating* kecocokan dari setiap alternatif pada setiap kriteria. Nilai X setiap alternatif (A_i) pada setiap kriteria (C_j) yang sudah ditentukan, dimana, $i=1,2,\dots,m$ dan $j=1,2,\dots,n$.

$$X = \begin{bmatrix} x_{11} & \dots & x_{1j} \\ \vdots & \ddots & \vdots \\ x_{i1} & \dots & x_{ij} \end{bmatrix} \quad (2-3)$$

- Melakukan normalisasi matrik keputusan dengan cara menghitung nilai *rating* kinerja ternormalisasi (r_{ij}) dari alternatif A_i pada kriteria C_j .

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\max_i x_{ij}}, & \text{jika } j \text{ atribut benefit} \\ \frac{\min_i x_{ij}}{x_{ij}}, & \text{jika } j \text{ atribut cost} \end{cases} \quad (2-4)$$

Keterangan:

r_{ij} = *rating* kinerja ternormalisasi

x_{ij} = *rating* kinerja dari alternatif A_i pada atribut C_j

- Hasil dari nilai *rating* kinerja ternormalisasi (r_{ij}) membentuk matrik ternormalisasi (R)

$$R = \begin{bmatrix} r_{11} & \dots & r_{1j} \\ \vdots & \ddots & \vdots \\ r_{i1} & \dots & r_{ij} \end{bmatrix} \quad (2-5)$$

- Hasil akhir nilai preferensi (V_i) diperoleh dari penjumlahan dari perkalian elemen baris matrik ternormalisasi (R) dengan bobot preferensi (W) yang bersesuaian elemen kolom matrik (W).

$$V_i = \sum_{j=1}^n W_j r_{ij} \quad (2-6)$$

Hasil perhitungan nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i merupakan alternatif terbaik (Usito, N.J, 2013).

BAB 3 METODOLOGI

Metodologi penelitian menjelaskan metode yang digunakan dalam pembuatan sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scraping* dan *fuzz-Simple Additive Weighting* (SAW). Tahapan metodologi penelitian ini meliputi studi literature, pengumpulan data, analisis kebutuhan, perancangan sistem, implementasi sistem, pengujian sistem dan kesimpulan.

3.1 Studi Literatur

Mempelajari literatur dari berbagai bidang ilmu yang berhubungan dengan pembuatan dan perancangan sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scraping* dan *fuzzy-Simple Additive Weighting* (SAW) diantaranya:

1. Pemograman dengan Bahasa PHP berbasis *Objek Oriented Programming* (OOP) dengan menggunakan *framework* Codeigniter 3.1
2. Pemodelan *database* MySQL dengan model *Fuzzy* untuk menyimpan data barang *online*
3. *Hypertext Markup Language* (HTML) Parsing menggunakan PHP
4. Sistem pendukung keputusan dengan metode *Simple Additive Weighting*
5. Rekayasa perangkat lunak
6. Data barang *online* dari situs *E-commerce* Indonesia

3.2 Pengumpulan Data

Sumber data didapatkan dari penelusuran atau *scrapping element* HTML yang ada pada 2 situs *E-commerce* yaitu Bukalapak dan Elevenia, yang telah menjadi batasan pada BAB I.

3.3 Analisis Kebutuhan

Analisa Kebutuhan dilakukan dengan menentukan apa saja yang dibutuhkan untuk membangun sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scrapping* dan *fuzzy-simple additive weighting*. Berikut ini kebutuhan yang digunakan untuk membangun sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scrapping* dan *fuzzy-simple additive weighting*:

1. Kebutuhan Hardware, meliputi:
 - Laptop dengan *processor* Intel® Core™ i7-4700HQ CPU @ 2.40GHz
 - Memory RAM 4.00 GB
2. Kebutuhan Software, meliputi:

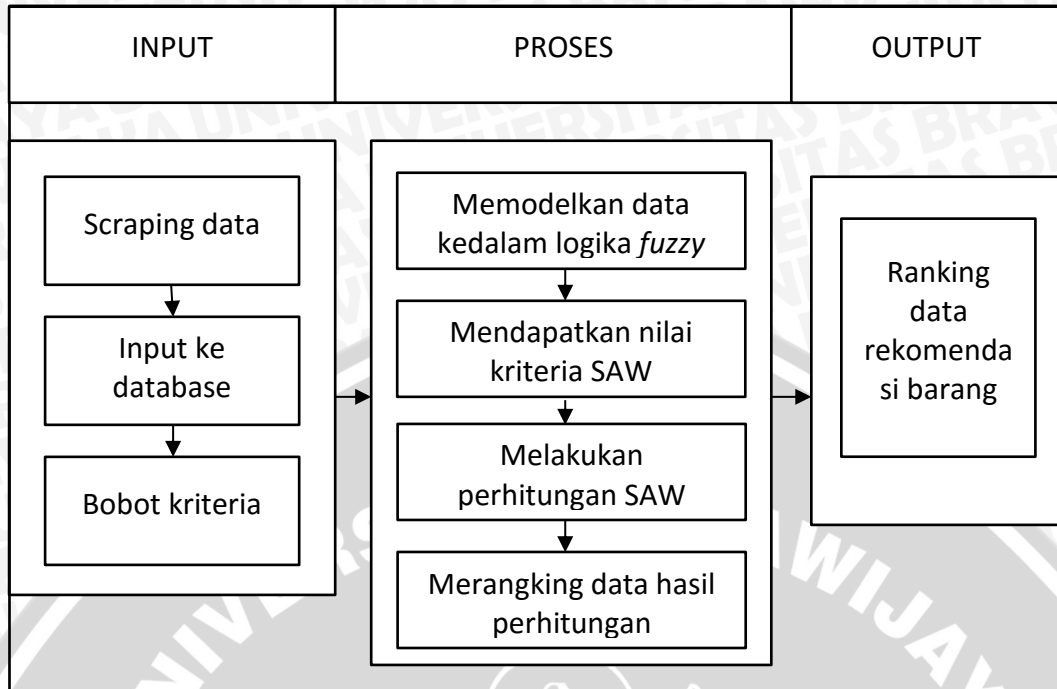
- *Operating System Windows 10 Pro 64 bit*
 - *XAMPP Server Version 3.1.0*
 - *Aplikasi PHP Storm dari JetBrains Version 10.5.0*
 - *Database MySQL Version 5.5.16*
 - *Framework CodeIgniter Version 3.1.0*
 - *Simple HTML DOM library Version 1.3.0*
 - *Materialize CSS library.*
 - *Jquery library Version 2.1.4*
3. Kebutuhan Data, meliputi:
- Data barang *online* dari situs *E-commerce* bukalapak dan elevenia.

3.4 Perancangan Sistem

Perancangan sistem dibuat mulai dari proses pengambilan data barang, penyimpanan data barang, pemrosesan data barang dan rekomendasi data barang, Hingga menghasilkan urutan rekomendasi barang. Hal ini bertujuan untuk mempermudah implementasi sistem.

3.4.1 Model Perancangan Sistem

Semua proses dapat dimodelkan mulai dari perancangan sistem, pencarian data, mengolah data, dan rekomendasi barang, dapat dilihat melalui Gambar 3.1.



Gambar 3.1 Model Perancangan Sistem

3.5 Implementasi Sistem

Implementasi sistem adalah fase membangun sistem yang mengacu pada perancangan sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scrapping* dan *fuzzy-simple additive weighting* dan menerapkan hal yang telah didapatkan dalam proses studi literatur. Fase-fase yang ada dalam implementasi antara lain:

- Implementasi tampilan (user interface), menggunakan software editor PHP Storm dari JetBrains
- Implementasi basis data, menggunakan DBMS MySQL pada *server localhost* (XAMPP).
- Implementasi *library simple HTML DOM*, untuk mendapatkan data dari situs *E-commerce*.
- Implementasi
- Implementasi algoritma dari metode *Fuzzy Simple Additive Weighting* (SAW) untuk perhitungan bobot kriteria menggunakan bahasa pemrograman php.
- Implementasi perangkingan data barang yang diurutkan berdasarkan ranking menggunakan *library javascript tablesorter*.

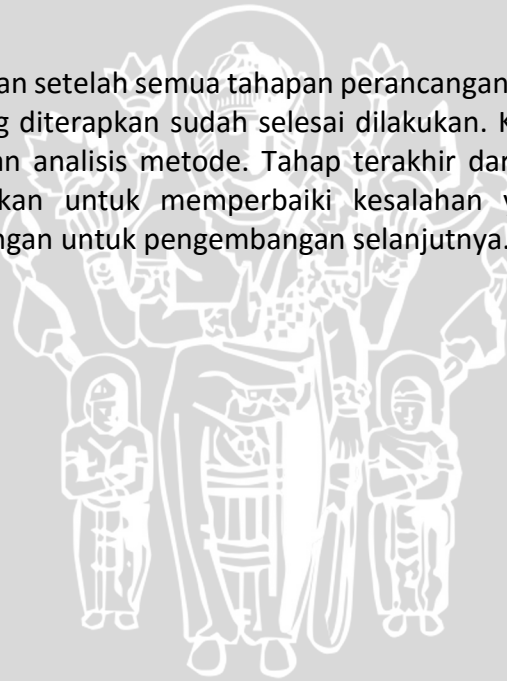
3.6 Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa aplikasi yang telah dibuat dapat berjalan sesuai dengan yang telah dirumuskan sebelumnya. Tujuan dari pengujian antara lain supaya dalam penggunaan sistem ini pengguna tidak menemui *error* atau *bug*. Pengujian yang dimaksud adalah:

1. Pengujian validasi sistem, yaitu pengujian dengan menggunakan *blackbox testing* untuk mengetahui kesesuaian antara hasil pengujian akurasi sistem pendukung keputusan yang terdapat pada perancangan dan implementasi.
2. Pengujian akurasi data hasil *output* sistem pendukung keputusan, yakni membandingkan hasil *output* sistem dengan hasil *output* perhitungan manualisasi.

3.7 Kesimpulan

Kesimpulan dilakukan setelah semua tahapan perancangan, implementasi dan pengujian metode yang diterapkan sudah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis metode. Tahap terakhir dari penulisan adalah saran yang dimaksudkan untuk memperbaiki kesalahan yang terjadi serta memberikan pertimbangan untuk pengembangan selanjutnya.



BAB 4 PERANCANGAN

Bab perancangan, akan menjelaskan beberapa langkah yang digunakan untuk mencapai target akhir penelitian ini, diantaranya, analisis kebutuhan, manajemen data, manajemen model, perancangan algoritma, dan antar muka pengguna. Dalam analisis kebutuhan perangkat lunak, terdapat tahapan identifikasi actor, daftar kebutuhan, diagram alir. Dalam manajemen data terdapat tahapan pemodelan dengan *Entity Relational Diagram* (ERD), selanjutnya terdapat penjelasan mengenai manajemen model, perancangan algoritma, dan perancangan antar muka pengguna.

4.1 Analisis Kebutuhan Perangkat Lunak

Proses analisa kebutuhan mengacu pada gambaran umum sistem. Proses analisis kebutuhan ini diawali dengan identifikasi aktor - aktor yang terlibat dalam sistem dan penjabaran tentang daftar kebutuhan. Analisis kebutuhan ini bertujuan untuk menggambarkan kebutuhan-kebutuhan yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan pengguna.

4.1.1 Identifikasi Aktor

Tahapan 4.1 bertujuan untuk mengidentifikasi aktor yang akan berinteraksi dengan sistem, jenis aktor, deskripsi aktor dan penjelasan detail mengenai aktor dapat dilihat pada Tabel 4.1

Tabel 4.1 Identifikasi Aktor

Aktor	Deskripsi Aktor	Keterangan
<i>Admin</i>	Pengatur Aplikasi	<i>Admin</i> adalah aktor yang mengatur jalannya aplikasi, seperti mengambil dan menghapus data barang
<i>User</i>	Pengguna Aplikasi	<i>User</i> adalah aktor yang mencari rekomendasi barang.

4.1.2 Daftar Kebutuhan

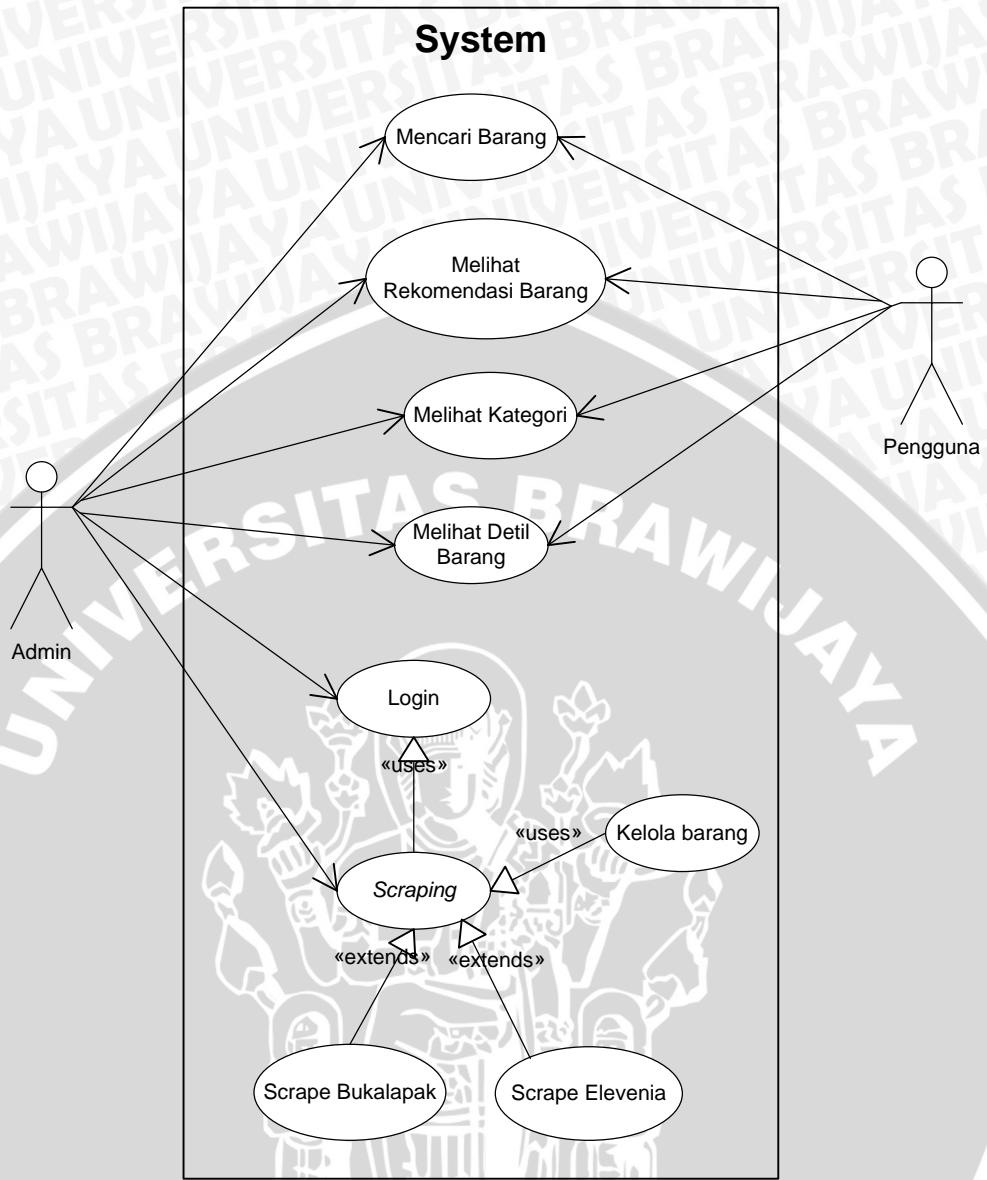
Tahapan 4.2 bertujuan untuk menganalisa dan merancang kebutuhan yang harus ada pada sistem, untuk lebih memudahkan kebutuhan disusun dengan menggunakan tabel. Kolom id digunakan untuk penomoran untuk setiap kebutuhan, kolom detail kebutuhan digunakan untuk penjelasan kebutuhan yang lebih detail, dan kolom nama aliran digunakan untuk penamaan yang lebih luas berdasarkan kebutuhan. Daftar kebutuhan sistem dijelaskan pada Tabel 4.2.

Tabel 4.2 Kebutuhan Fungsional

Id	Detil Kebutuhan	Nama Aliran
001	Sistem mampu untuk mengambil atau mendapatkan data barang dari situs <i>E-commerce</i>	<i>Scrape</i>
002	Sistem mampu menampilkan detail informasi barang	<i>Detail</i>
003	Sistem mampu untuk merekomendasikan barang terbaik dari berbagai kriteria berdasarkan kata kunci	Rekomendasi
004	Sistem mampu untuk mengurutkan barang yang direkomendasikan dari yang terbaik	Urutan
005	Sistem mampu untuk menghapus barang yang tidak relevan.	Hapus
006	Sistem mampu untuk melakukan pencarian barang berdasarkan kata kunci yang di masukkan	Pencarian
007	Sistem mampu untuk menampilkan hasil pencarian	Hasil
008	Sistem mampu untuk menampilkan kategori barang	Kategori

4.1.3 Diagram Use Case

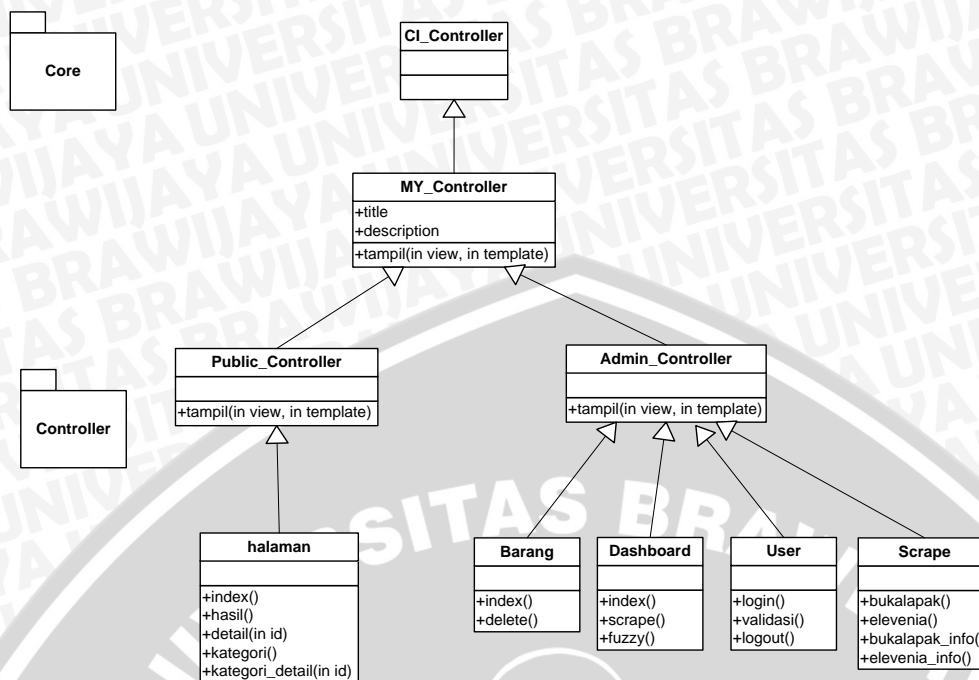
Use case digunakan untuk memodelkan dan menyatakan unit fungsi/layanan yang disediakan oleh sistem atau bagian sistem atau subsistem, kepada pemakai, pada tahapan ini akan digambarkan *use case* sistem yang akan dirancang. Untuk lebih jelas lihat Gambar 4.1



Gambar 4.1 Use Case Diagram

4.1.4 Diagram Class

Sistem akan dirancang menggunakan konsep *Object Oriented Programming* (OOP) maka sistem membutuhkan rancangan *class diagram* yang berfungsi untuk merancang *class-class* yang terlibat didalam sistem. Untuk penjelasan lihat Gambar 4.2.



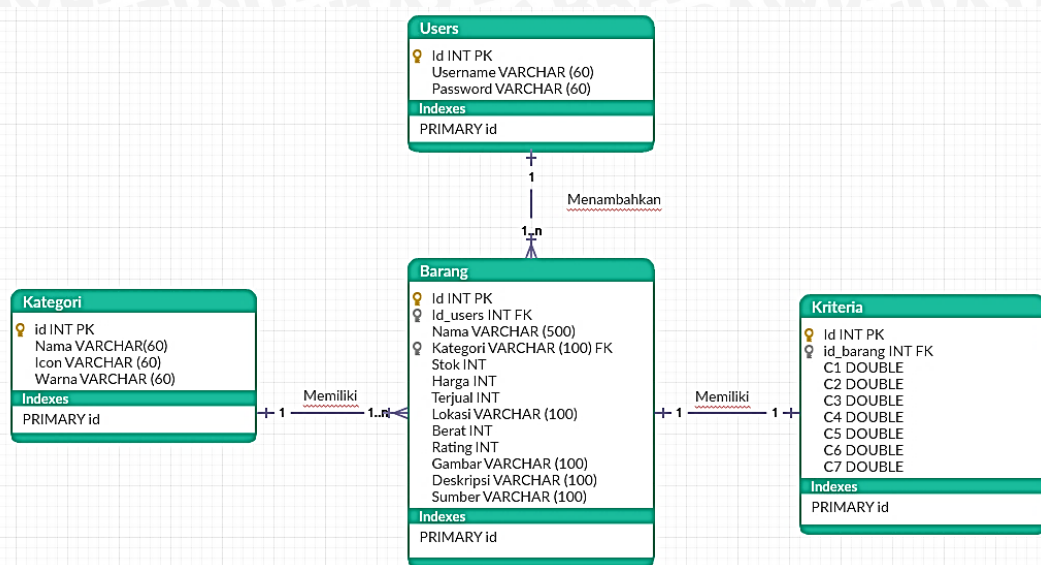
Gambar 4.2 Diagram Class

4.2 Manajemen Data

Proses manajemen data merupakan pengorganisasian data untuk lebih mudah untuk diterapkan kedalam sebuah sistem. Adapun pembahasan dari manajemen data meliputi *Entity Relational Diagram* (ERD), tabel basis data yang akan dirancang, basis pengetahuan, pembobotan dan kriteria, serta normalisasi data.

4.2.1 Entity Relationship Diagram

Untuk memudahkan perancangan *database* dibutuhkan sebuah diagram relasi yang memetakan antara entitas, atribut dan relasi antar entitas, *Entity Relational Diagram* (ERD) adalah sebuah konsep yang mendeskripsikan hubungan antara penyimpanan *database* dan persepsi dari sebuah objek di dunia nyata, yang terdiri dari sekumpulan objek yang disebut sebagai *entity* dan hubungan atau relasi antara objek-objek tersebut. *Entity Relationship Diagram* dari sistem pendukung keputusan ini dapat dilihat pada Gambar 4.3.



Gambar 4.3 Entity Relationship Diagram

4.2.2 Tabel Basis Data

Sistem yang memiliki kompleksitas tinggi harus memiliki basis data untuk menyimpan sebagian atau seluruh data, pada tahapan ini perancangan tabel basis data dibuat berdasarkan entitas dan fungsi yang telah dirancang pada *Entity Relationship Diagram* (ERD), terdiri dari tabel SK_Barang, SK_Banding, SK_Kategori dan SK_Kriteria.

4.2.2.1 Tabel Data Barang

Data informasi barang disimpan dalam tabel SK_Barang, adapun atribut atau kolom yang dimiliki oleh tabel SK_Barang adalah id, nama, kategori, harga, stok, terjual, berat, *rating*, gambar, deskripsi dan sumber. Tabel SK_Barang untuk lebih jelasnya dapat dilihat pada Tabel 4.3.

Tabel 4.3 Tabel SK_Barang

Nama	Tipe	Panjang data	Keterangan
Id	Int	11	Auto Increment
Nama	Varchar	500	-
Kategori	Varchar	500	-
Harga	Int	11	-
Stok	Int	11	-
Terjual	Int	11	-
Lokasi	Varchar	100	-
Berat	Int	11	-
<i>Rating</i>	Int	11	-

Gambar	Varchar	500	-
Deskripsi	Varchar	500	-
Sumber	Varchar	500	-

4.2.2.2 Tabel Kategori barang

Data informasi barang berdasarkan kategorinya disimpan dalam tabel SK_Kategori, adapun atribut atau kolom yang dimiliki oleh tabel SK_Kategori adalah id, nama, *icon*, warna. Tabel SK_Kategori untuk lebih jelasnya dapat dilihat pada Tabel 4.4.

Tabel 4.4 Tabel SK_Kategori

Nama	Tipe	Panjang data	Keterangan
Id	Int	11	Primary Key
Nama	Varchar	60	-
Icon	Varchar	60	-
Warna	Varchar	60	-

4.2.2.3 Tabel Kriteria barang

Data informasi barang disimpan dalam tabel SK_Kriteria, adapun atribut atau kolom yang dimiliki oleh tabel SK_Kriteria adalah id, C1, C2, C3, C4, C5, C6, C7. Tabel SK_Kriteria untuk lebih jelasnya dapat dilihat pada Tabel 4.5.

Tabel 4.5 Tabel SK_Kriteria

Nama	Tipe	Panjang data	Keterangan
Id	int	11	Primary Key
C1	double	60	-
C2	double	60	-
C3	double	60	-
C4	double	60	-
C5	double	60	-
C6	double	60	-
C7	double	60	-

4.2.3 Basis Pengetahuan

Perancangan basis pengetahuan bertujuan mendukung perancangan lainnya untuk memberikan sejumlah pengetahuan dan proses intelegensi dalam pengambilan keputusan. Dalam tahapan ini dibutuhkan pengetahuan dari seorang ahli atau berdasarkan pengamatan lapangan untuk menentukan bobot kriteria yang dibutuhkan oleh sistem untuk melakukan penilaian. Nilai bobot sub-kriteria ditentukan dengan menggunakan metode *Rank Order Centroid* (ROC). Nilai dari bobot kriteria dan sub-kriteria tersebut akan diproses dengan metode *Simple Additive Weighting*.

4.2.3.1 Himpunan variabel kriteria

Himpunan variabel kriteria yang terdapat pada sistem pendukung keputusan ini terdiri dari:

- A1 ... An = Alternatif barang
- C1 = Harga barang
- C2 = Berat barang
- C3 = Stok barang
- C4 = Lokasi pengiriman barang
- C5 = Panjang teks deskripsi barang
- C6 = *Rating* barang
- C7 = Penjualan unit barang

4.2.3.2 Data dan kriteria barang

Data barang beserta nilai disetiap kriterianya yang akan diproses dengan metode *Simple Additive Weighting*, ditunjukkan pada Tabel 4.6.

Tabel 4.6 Kriteria dan penilaian barang

Alternatif	Kriteria						
	C1	C2	C3	C4	C5	C6	C7
A1	3278000	200	1	Depok	486	100	1
A2	4850000	250	1	Jakarta Pusat	53	100	1
A3	3799000	700	10	Jakarta Barat	455	99	2
A4	2699000	600	4	Jakarta Barat	455	99	1
A5	135000	150	6	Jakarta Timur	476	100	3
A6	6100000	133	7	Jakarta Barat	490	100	2
A7	2150000	500	10	Jakarta Barat	480	93	1

A9	3599000	600	5	Jakarta Barat	455	99	6
A10	2399000	500	6	Jakarta Barat	469	99	194
A11	2450000	500	4	Jakarta Selatan	478	95	17
A12	2390000	500	4	Jakarta Selatan	477	95	7
A13	2350000	500	100	Jakarta Utara	464	98	41
A14	2150000	500	50	Jakarta Utara	463	98	40
A15	2250000	500	5	Jakarta Selatan	474	95	4
A16	2150000	400	100	Jakarta Timur	476	95	28
A17	3000000	10	1	Sleman	427	91	1
A18	2750000	500	10	Jakarta Timur	471	95	13
A19	2860000	500	6	Jakarta Selatan	340	95	12
A20	2399000	500	10	Jakarta Barat	469	99	12

4.2.4 Bobot Preferensi dan Kriteria Data

Penentuan nilai bobot ditentukan berdasarkan ahli atau pengamatan spesifik terhadap satu kasus, pada sistem pendukung keputusan ini nilai bobot ditentukan berdasarkan pengamatan pada situs *E-commerce*, dan wawancara terhadap beberapa pengguna *E-commerce*, atau ditentukan oleh pengguna. Adapun nilai bobot preferensi yang diterapkan secara *default* pada sistem pendukung keputusan ini yaitu, harga barang 25%, berat barang 7%, lokasi pengiriman 8%, deskripsi barang 13%, *rating/review* 15%, stok barang 7%, penjualan unit barang 25%, pada sistem nilai bobot preferensi sepenuhnya diatur oleh pengguna.

1. Range nilai sub-kriteria harga barang

Kriteria harga barang terbagi atas 5 sub-kriteria yaitu sangat murah, murah, sedang, mahal, sangat mahal. Dan penilaian menggunakan rumasan *Range Order Centroid* (ROC) dalam menetapkan variabel *fuzzy*. Untuk lebih jelas dapat dilihat pada Tabel 4.7.

Tabel 4.7 Parameter kriteria harga

Harga (C1)	Nilai
Sangat Murah <= Rp. 50.000	0.04
Murah = 50.000 – 500.000	0.09

Sedang = 500.000 – 5.000.000	0.156667
Mahal = 5.000.000 – 10.000.000	0.256667
Sangat Mahal >= Rp. 10.000.000	0.456667

2. Range nilai sub-kriteria berat barang

Kriteria berat barang terbagi atas 3 sub-kriteria yaitu ringan, sedang, dan berat. Penilaian berdasarkan rumusan *Range Order Centroid* (ROC) untuk menetapkan variabel *fuzzy*. Untuk lebih jelas dapat dilihat pada Tabel 4.8.

Tabel 4.8 Parameter kriteria berat barang

Berat(gram) (C2)	Nilai
Ringan <= 100	0.111111
Sedang = 100 – 2000	0.277778
Berat => 2.000	0.611111

3. Range nilai sub-kriteria berat barang

Kriteria stok barang terbagi atas 5 sub-kriteria yaitu sangat sedikit, sedikit, sedang, banyak, sangat banyak. Penilaian berdasarkan rumusan *Range Order Centroid* (ROC) untuk menetapkan variabel *fuzzy*. Untuk lebih jelas dapat dilihat pada Tabel 4.10.

Tabel 4.9 Parameter kriteria stok barang

Stok (C3)	Nilai
Sangat Sedikit <= 5	0.04
Sedikit = 5-20	0.09
Sedang = 20-100	0.156667
Banyak = 100-500	0.256667
Sangat Banyak >= 500	0.456667

4. Range nilai sub-kriteria lokasi pengiriman barang

Kriteria lokasi pengiriman terbagi atas 3 sub-kriteria yaitu berada dalam kota Jakarta, Pulau Jawa, luar Pulau Jawa. Dan penilaian berdasarkan range order centroid untuk menetapkan variabel *fuzzy*. Untuk lebih jelas dapat dilihat pada Tabel 4.11.

Tabel 4.10 Parameter kriteria lokasi pengiriman barang

Lokasi Pengiriman (C4)	Nilai
------------------------	-------

Jakarta	0.111111
Pulau Jawa	0.277778
Luar Pulau Jawa	0.611111

5. Range nilai sub-kriteria deskripsi barang

Kriteria harga barang terbagi atas 5 sub-kriteria yaitu sangat murah, murah, sedang, mahal, sangat mahal. Penilaian berdasarkan rumusan *Range Order Centroid* (ROC) untuk menetapkan variabel *fuzzy*. Range nilai sub-kriteria deskripsi barang dapat dilihat pada Tabel 4.12.

Tabel 4.11 Parameter kriteria deskripsi barang

Deskripsi (C5)	Nilai
Tidak Jelas \leq 50 huruf	0.0625
Kurang Jelas = 50-200 huruf	0.145833
Jelas = 200-1000 huruf	0.270833
Sangat Jelas \geq 1000 huruf	0.520833

6. Range nilai sub-kriteria lokasi pengiriman barang

Kriteria harga barang terbagi atas 5 sub-kriteria yaitu sangat murah, murah, sedang, mahal, sangat mahal. Penilaian berdasarkan rumusan *Range Order Centroid* (ROC) untuk menetapkan variabel *fuzzy*. Range nilai sub-kriteria lokasi pengiriman barang dapat dilihat pada Tabel 4.13.

Tabel 4.12 Parameter kriteria *rating/review*

<i>Rating/Review</i> (C6)	Nilai
Reputasi worst = 0 - 20	0.04
Reputasi bad = 20 - 40	0.09
Reputasi average = 40 - 60	0.156667
Reputasi good = 60 - 80	0.256667
Reputasi very-good = 80 - 100	0.456667

7. Range nilai sub-kriteria penjualan unit barang

Kriteria harga barang terbagi atas 5 sub-kriteria yaitu sangat murah, murah, sedang, mahal, sangat mahal. Penilaian berdasarkan rumusan *Range Order Centroid* (ROC) untuk menetapkan variabel *fuzzy*. Range nilai sub-kriteria penjualan unit barang dapat dilihat pada Tabel 4.14.

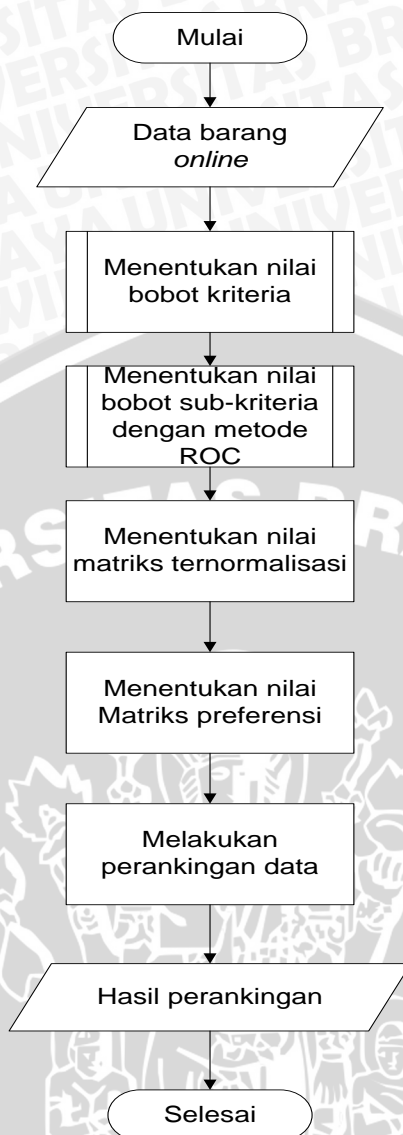
Tabel 4.13 Parameter kriteria penjualan unit barang

Penjualan (C7)	Nilai
Sangat Sedikit ≤ 2	0.04
Sedikit = 2 – 10	0.09
Sedang = 10 – 100	0.156667
Banyak = 100 – 500	0.256667
Sangat Banyak ≥ 500	0.456667

4.3 Perancangan Diagram Alir dan Algoritma

Algoritma berfungsi untuk memudahkan pembuatan program aplikasi, dengan adanya algoritma, seorang *programmer* lebih terarah untuk menulis kode sumber, yang selanjutnya akan menjadi aplikasi. Dalam perancangan sistem pendukung keputusan, algoritma digunakan untuk melakukan proses pendukung keputusan.

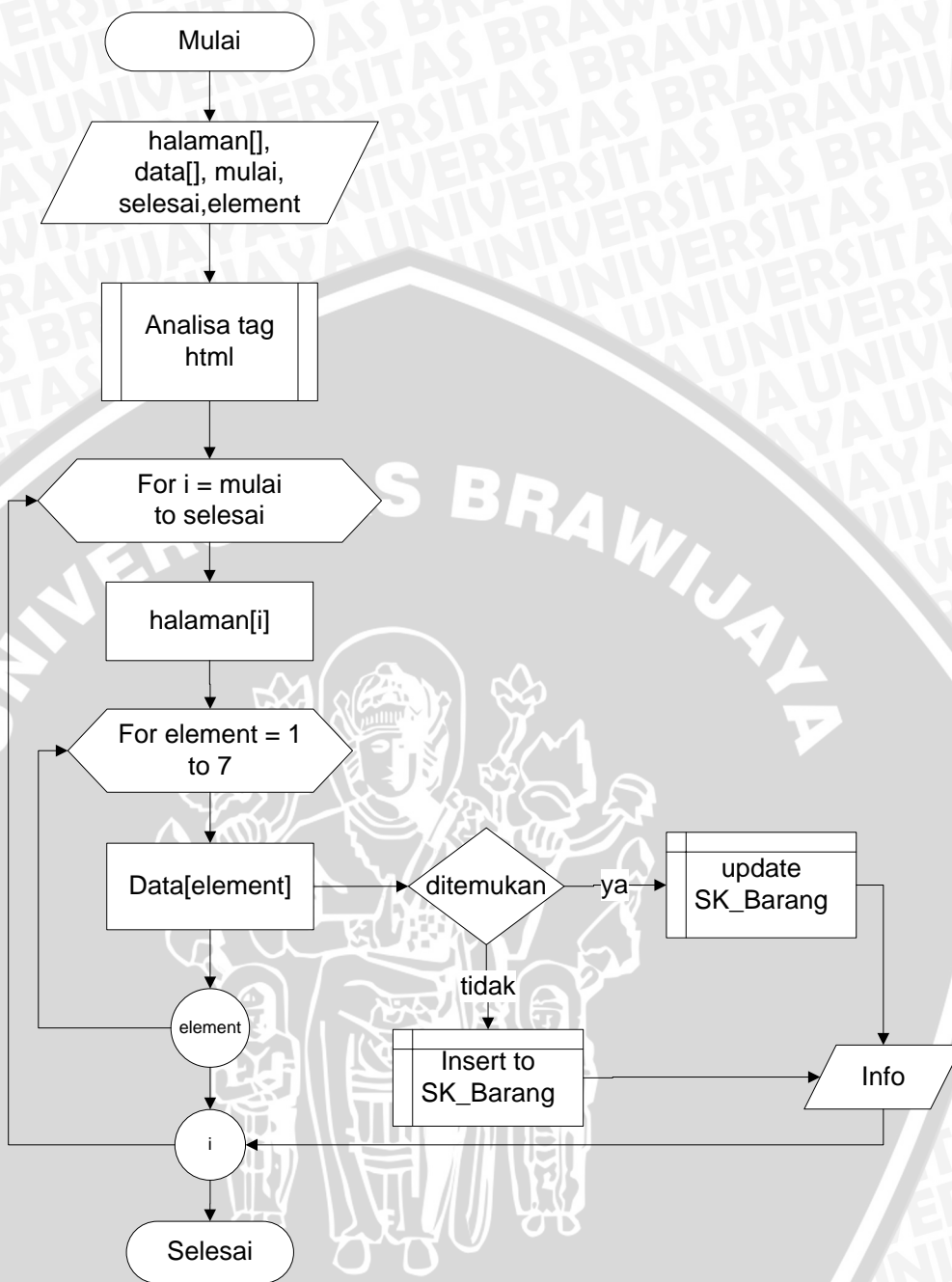
Algoritma metode *simple additive weighting* digunakan untuk menghitung nilai proses keseluruhan, yaitu melakukan perhitungan bobot kriteria, menghitung nilai matriks ternormalisasi setiap matriks. Menghitung matriks preferensi setiap alternative, dan melakukan perankingan. Adapun *flowchart* algoritma dari metode *simple additive weighting* dapat dilihat pada Gambar 4.4.



Gambar 4.4 Diagram alir metode *Simple Additive Weighting*

4.3.1 Algoritma *Scraping* data barang

Data yang tersebar pada situs *E-commerce* dapat diambil dengan metode *web scraping*, *scraping* data barang dilakukan dengan cara mengambil informasi yang terdapat pada *tag HTML*. Diagram alir *web scraping* dapat dilihat pada Gambar 4.5.



Gambar 4.5 Diagram alir *scraping* situs *E-commerce*

Nama: Algoritma *Scraping* Data Barang

Deskripsi: mendapatkan data barang pada situs *e-commerce* Bukalapak dan Elevenia

Deklarasi:

- Array halaman[] merupakan halaman yang akan diambil data/informasinya
- Array data[] merupakan variabel untuk menyimpan data yang terdapat pada halaman
- Mulai adalah variabel yang menunjukkan halaman pertama yang akan diambil datanya
- Selesai adalah variabel yang menunjukkan halaman terakhir yang akan diambil datanya
- Element adalah variabel yang menunjukkan data sesuai kriteria barang

Input: halaman[], data[], mulai, selesai, element

Proses:

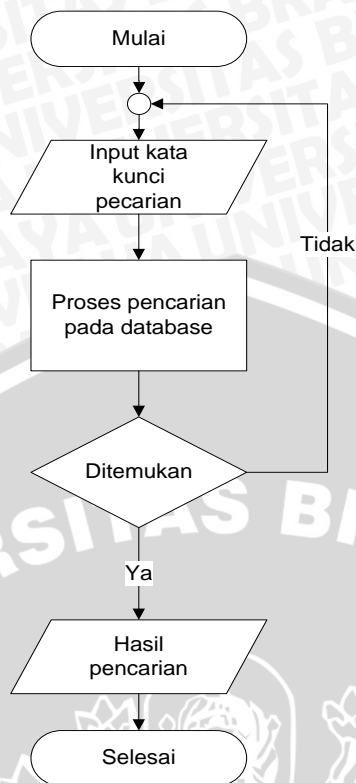
1. melakukan perulangan pada halaman html yang akan diambil datanya dimulai dari $i = \text{halaman awal}$ hingga $i = \text{halaman akhir}$
2. melakukan perulangan pada data setiap element/tag html yang sesuai dengan kriteria barang
3. Menyimpan informasi barang pada variabel array data[]
4. Menyimpan data yang didapatkan kedalam database
5. Jika data sudah ada maka data akan *update*, jika data belum ada maka data akan *insert*

Output: laporan info barang yang berhasil dimasukkan dan tidak

Gambar 4.6 Algoritma *scraping* data barang

4.3.2 Algoritma Pencarian Data

Pencarian data digunakan untuk mencari barang spesifik untuk dilakukan perankingan. Perankingan barang akan dilakukan dengan dua cara yaitu dengan pencarian kata kunci dan perankingan berdasarkan kategori. Lihat Gambar 4.7.



Gambar 4.7 Diagram alir pencarian data barang

Selain penjelasan dengan menggunakan diagram alir juga terdapat penulisan algoritma untuk memudahkan penulisan kode sumber. Berikut algoritma pencarian data dijelaskan pada Gambar 4.8.

Nama: Algoritma Pencarian Data

Deskripsi: melakukan pencarian data barang sesuai dengan keyword yang dicari

Deklarasi:

- q merupakan kata kunci pencarian yang digunakan mencari barang

Input: q, cat

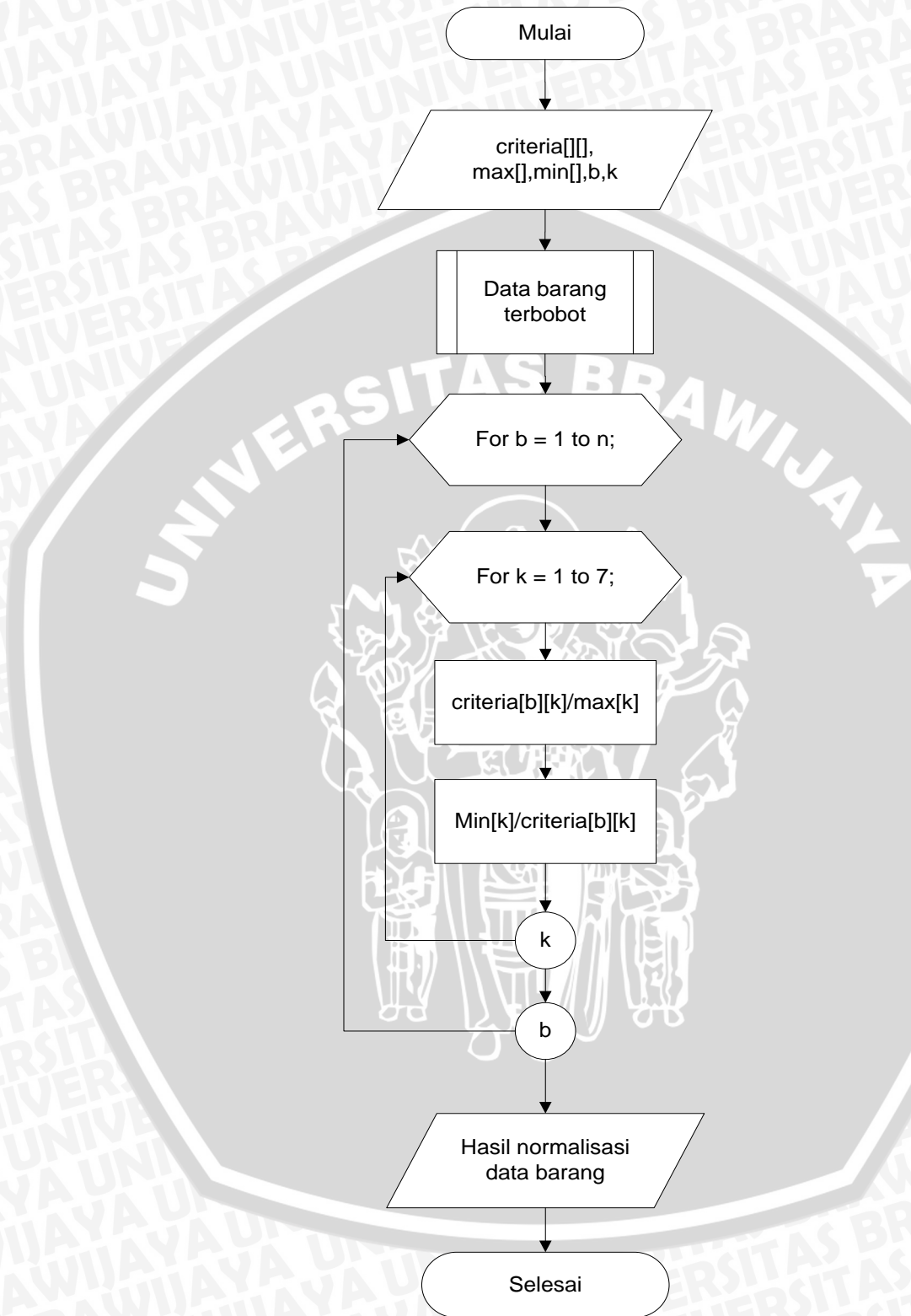
Proses:

1. Pencarian data serupa pada basis data, berdasarkan keyword(q) yang dimasukkan
2. Pencarian data spesifik berdasarkan kategori yang dipilih

Output: Hasil pencarian data

Gambar 4.8 Algoritma pencarian data barang

4.3.3 Algoritma Normalisasi Data



Gambar 4.9 Diagram alir normalisasi

Selain penjelasan dengan menggunakan diagram alir pada Gambar 4.9 juga terdapat algoritma untuk memudahkan penulisan kode sumber. Berikut algoritma pencarian data dijelaskan pada Gambar 4.10.

Nama Algoritma: Perhitungan Normalisasi Matriks

Deskripsi: menghitung nilai normalisasi matriks dari setiap alternatif data barang

Deklarasi:

- Double Array criteria[][] merupakan nilai alternatif dari setiap kriteria barang yang telah dimasukkan dalam database
- Double Array max[] nilai maksimum dari semua nilai alternative pada criteria ke n
- Double Array min[] nilai minimum dari semua nilai alternative pada criteria ke n
- B merupakan variabel untuk baris, k variabel untuk kolom

Masukan:

- criteria[[]], max[], min[], b, k

Pre-Proses:

- Mengubah nilai data mentah dan memasukkan kedalam database berdasarkan kondisi nilai

Proses:

1. Lakukan perulangan untuk mendapatkan nilai kriteria disetiap alternatif.
2. Kemudian, hasil dari penjumlahan tersebut disimpan pada variabel array max[] untuk benefit dan min[] untuk cost.
3. Menampilkan nilai normalisasi matriks setiap alternatif

Keluaran : Hasil perhitungan normalisasi matriks alternatif

Gambar 4.10 Algoritma Normalisasi

Dari Algoritma 4.10, secara matematis proses normalisasi matriks keputusan dilakukan dengan cara menghitung nilai *rating* kinerja ternormalisasi (r_{ij}) dari alternatif A_i pada kriteria C_j .

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{Max}_i x_{ij}}, & \text{jika } j \text{ atribut benefit} \\ \frac{\text{Min}_i x_{ij}}{x_{ij}}, & \text{jika } j \text{ atribut cost} \end{cases}$$

$$r_{11} = \frac{\text{Min}(0.15,0.15,0.15,0.15,0.15,0.25,0.15,0.15,0.15,0.15,0.15,0.15,0.15,0.15,0.15)}{0.15} = 1.0$$

$$r_{12} = \frac{\text{Min}(0.27,0.27,0.27,0.27,0.27,0.27,0.27,0.27,0.27,0.27,0.27,0.27,0.27,0.27,0.11)}{0.11} = 0.4$$

.....

$$r_{2020} = \frac{0.15}{\text{Max}(0.04,0.04,0.04,0.04,0.09,0.04,0.04,0.04,0.09,0.25,0.15,0.09,0.15,0.15,0.09,0.15)} = 0.61$$

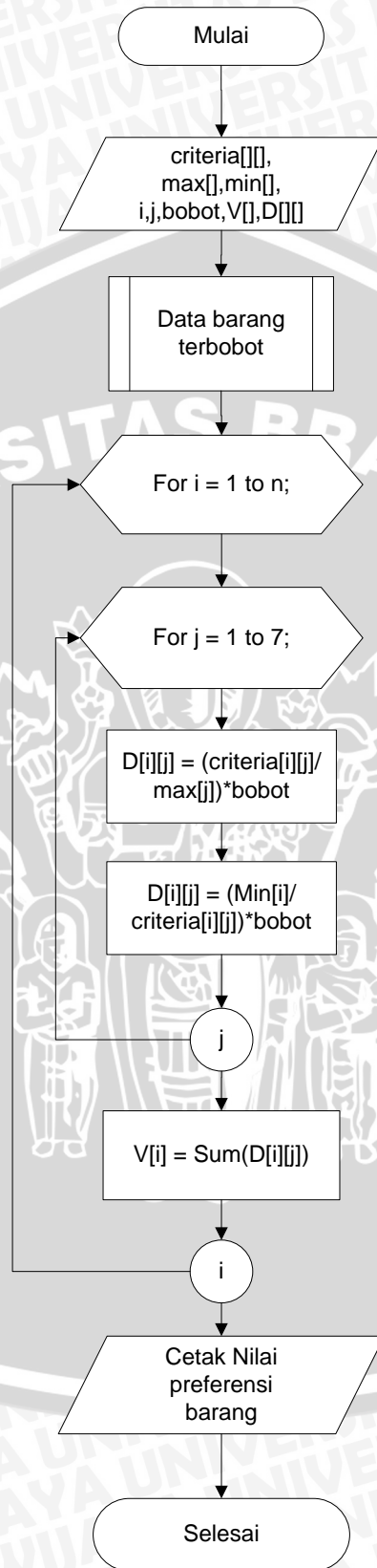
Sehingga didapatkan hasil data barang yang telah ternormalisasi, dan dapat dilihat pada Tabel 4.15

Tabel 4.14 Normalisasi data

Alternatif	Kriteria						
	C1	C2	C3	C4	C5	C6	C7
A1	1.0000	0.4000	0.2553	0.4000	1.0000	1.0000	0.1558
A2	1.0000	0.4000	0.2553	1.0000	0.5385	1.0000	0.1558
A3	1.0000	0.4000	0.5745	1.0000	1.0000	1.0000	0.1558
A4	1.0000	0.4000	0.2553	1.0000	1.0000	1.0000	0.1558
A5	1.0000	0.4000	0.5745	1.0000	1.0000	1.0000	0.3506
A6	0.6104	0.4000	0.5745	1.0000	1.0000	1.0000	0.1558
A7	1.0000	0.4000	0.5745	1.0000	1.0000	1.0000	0.1558
A9	1.0000	0.4000	0.5745	1.0000	1.0000	1.0000	0.1558
A10	1.0000	0.4000	0.2553	1.0000	1.0000	1.0000	0.3506
A11	1.0000	0.4000	0.5745	1.0000	1.0000	1.0000	1.0000
A12	1.0000	0.4000	0.2553	1.0000	1.0000	1.0000	0.6104
A13	1.0000	0.4000	0.2553	1.0000	1.0000	1.0000	0.3506
A14	1.0000	0.4000	1.0000	1.0000	1.0000	1.0000	0.6104
A15	1.0000	0.4000	1.0000	1.0000	1.0000	1.0000	0.6104
A16	1.0000	0.4000	0.2553	1.0000	1.0000	1.0000	0.3506
A17	1.0000	0.4000	1.0000	1.0000	1.0000	1.0000	0.6104
A18	1.0000	1.0000	0.2553	0.4000	1.0000	1.0000	0.1558
A19	1.0000	0.4000	0.5745	1.0000	1.0000	1.0000	0.6104
A20	1.0000	0.4000	0.5745	1.0000	1.0000	1.0000	0.6104



4.3.4 Algoritma Preferensi Data



Gambar 4.11 Diagram alir nilai preferensi

Selain penjelasan dengan menggunakan diagram alir pada Gambar 4.11 juga terdapat algoritma untuk memudahkan penulisan kode sumber. Berikut algoritma nilai preferensi data yang dijelaskan pada Gambar 4.12

Nama: Algoritma Preferensi Data

Deskripsi: perhitungan nilai preferensi data barang, lalu menjumlahkannya untuk mendapatkan nilai preferensi setiap alternatif

Deklarasi:

- criteria[][] merupakan nilai kriteria alternatif dari setiap kriteria barang
- D[][] merupakan nilai perhitungan criteria[][] dikalikan dengan bobot
- v[] merupakan jumlah nilai preferensi setiap alternatif
- bobot merupakan nilai bobot yang ditentukan oleh pemberi keputusan
- i merupakan variabel untuk baris, j variabel untuk kolom

Masukkan: criteria[][],,D[][],vi[],bobot,i,j

Proses:

1. Melakukan proses perkalian antara nilai kriteria ternormalisasi, dengan nilai bobot yang telah ditentukan pendukung keputusan.
2. Menjumlahkan nilai preferensi setiap alternative disemua kriteria
3. Mengurutkan data dari terbesar hingga terkecil

Keluaran: Hasil nilai preferensi

Gambar 4.12 Algoritma Preferensi

Hasil dari data barang yang telah dilakukan perhitungan nilai preferensi dapat dilihat pada Tabel 4.16.

Tabel 4.15 Nilai preferensi setiap alternatif

Alternatif	Kriteria						
	C1	C2	C3	C4	C5	C6	C7
A1	0.2500	0.0280	0.0179	0.0320	0.1300	0.1500	0.0390
A2	0.2500	0.0280	0.0179	0.0800	0.0700	0.1500	0.0390
A3	0.2500	0.0280	0.0402	0.0800	0.1300	0.1500	0.0390
A4	0.2500	0.0280	0.0179	0.0800	0.1300	0.1500	0.0390
A5	0.2500	0.0280	0.0402	0.0800	0.1300	0.1500	0.0877

A6	0.1526	0.0280	0.0402	0.0800	0.1300	0.1500	0.0390
A7	0.2500	0.0280	0.0402	0.0800	0.1300	0.1500	0.0390
A9	0.2500	0.0280	0.0402	0.0800	0.1300	0.1500	0.0390
A10	0.2500	0.0280	0.0179	0.0800	0.1300	0.1500	0.0877
A11	0.2500	0.0280	0.0402	0.0800	0.1300	0.1500	0.2500
A12	0.2500	0.0280	0.0179	0.0800	0.1300	0.1500	0.1526
A13	0.2500	0.0280	0.0179	0.0800	0.1300	0.1500	0.0877
A14	0.2500	0.0280	0.0700	0.0800	0.1300	0.1500	0.1526
A15	0.2500	0.0280	0.0700	0.0800	0.1300	0.1500	0.1526
A16	0.2500	0.0280	0.0179	0.0800	0.1300	0.1500	0.0877
A17	0.2500	0.0280	0.0700	0.0800	0.1300	0.1500	0.1526
A18	0.2500	0.0700	0.0179	0.0320	0.1300	0.1500	0.0390
A19	0.2500	0.0280	0.0402	0.0800	0.1300	0.1500	0.1526
A20	0.2500	0.0280	0.0402	0.0800	0.1300	0.1500	0.1526

4.3.4.1 Perankingan data berdasarkan nilai preferensi

Nama: Algoritma Perankingan Data

Deskripsi: melakukan perankingan data, data hasil perhitungan total nilai matriks preferensi setiap alternative.

Deklarasi:

- Barang[] merupakan data dari hasil nilai prefensi

Masukkan: barang[]

Proses:

1. Urutkan barang berdasarkan nilai prefensi dengan *library tablesorter*

Keluaran: Hasil perankingan barang

Total nilai preferensi masing-masing alternatif dapat dilihat pada Table 4.17.

Tabel 4.16 Nilai total preferensi alternatif (Vi) sebelum diranking

Alternatif	Nilai preferensi
A1	0.6468
A2	0.6348
A3	0.7172
A4	0.6948
A5	0.7659



A6	0.6198
A7	0.7172
A8	0.7172
A9	0.7435
A10	0.9282
A11	0.8085
A12	0.7435
A13	0.8608
A14	0.8606
A15	0.7435
A16	0.8606
A17	0.6888
A18	0.8308
A19	0.8308
A20	0.8308

Perankingan alternatif berdasarkan nilai bobot prefesensinya yang diurutkan dari bilangan terbesar hingga terkecil, dapat dilihat pada Tabel 4.18.

Tabel 4.17 Nilai total preferensi alternative (Vi) setelah diranking

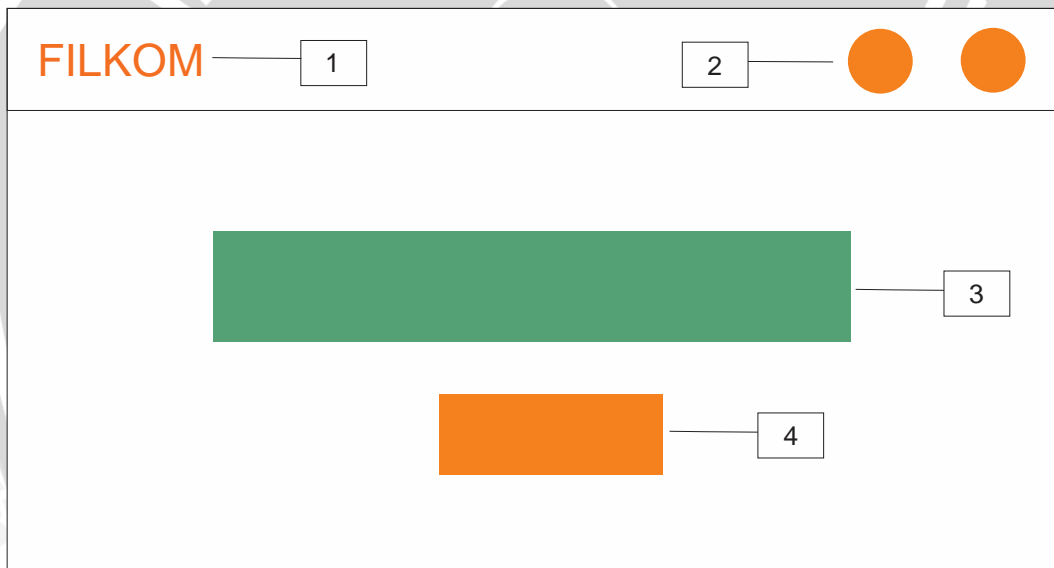
Alternatif	Total nilai	Ranking
A10	0.9282	1
A13	0.8608	2
A14	0.8606	3
A16	0.8606	4
A18	0.8308	5
A19	0.8308	6
A20	0.8308	7
A11	0.8085	8
A5	0.7659	9
A9	0.7435	10
A12	0.7435	11
A15	0.7435	12
A3	0.7172	13
A7	0.7172	14
A8	0.7172	15
A4	0.6948	16
A17	0.6888	17
A1	0.6468	18
A2	0.6348	19
A6	0.6198	20

4.4 Antar Muka Pengguna

Pengguna memerintahkan sistem melalui subsistem-subsistem antarmuka pengguna. Oleh karena itu, sistem ini harus menyediakan antarmuka yang memiliki pengalaman user (*User Experience*) yang baik. Perancangan antarmuka ini akan dijelaskan pada perancangan antarmuka pengguna. Perancangan antarmuka ini memiliki 5 halaman utama. Halaman tersebut meliputi halaman utama pencarian, halaman hasil pencarian, halaman detail, halaman pengaturan, dan halaman ranking. Berikut ini adalah perancangan antar muka sistem yang akan dibangun.

4.4.1 Halaman Utama Pencarian

Halaman utama yang akan dirancang berbentuk simple responsive, halaman utama memiliki 4 komponen penyusun halaman, yaitu logo FILKOM sebagai identitas, menu sebagai navigasi untuk menuju halaman kategori dan halaman banding, kolom pencarian, dan tombol kategori untuk menuju halaman kategori. Visualisasi halaman dapat dilihat pada Gambar 4.13.



Gambar 4.13 Halaman Home

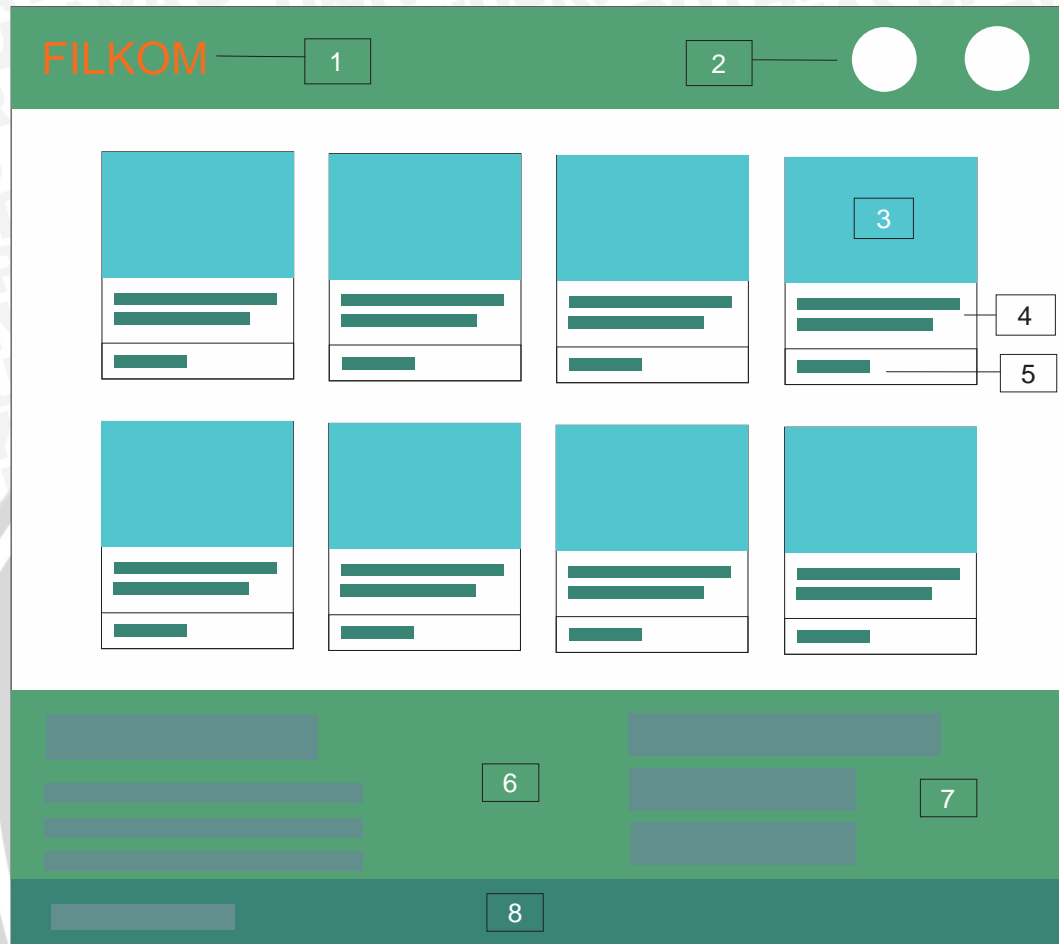
Keterangan Gambar 4.13:

1. Logo FILKOM sebagai identitas sistem
2. Menu kategori dan aktivasi rekomendasi
3. Kolom pencarian untuk melakukan pencarian
4. Tombol kategori untuk menuju halaman kategori

4.4.2 Halaman Hasil Pencarian Barang

Halaman hasil pencarian barang yang akan dirancang memiliki 8 komponen penyusun halaman, yaitu logo FILKOM sebagai identitas, menu *top-navbar* sebagai

navigasi untuk menuju halaman kategori, gambar, nama dan harga barang, pernyataan dan sumber pada bagian *footer*. Visualisasi halaman dapat dilihat pada Gambar 4.14.



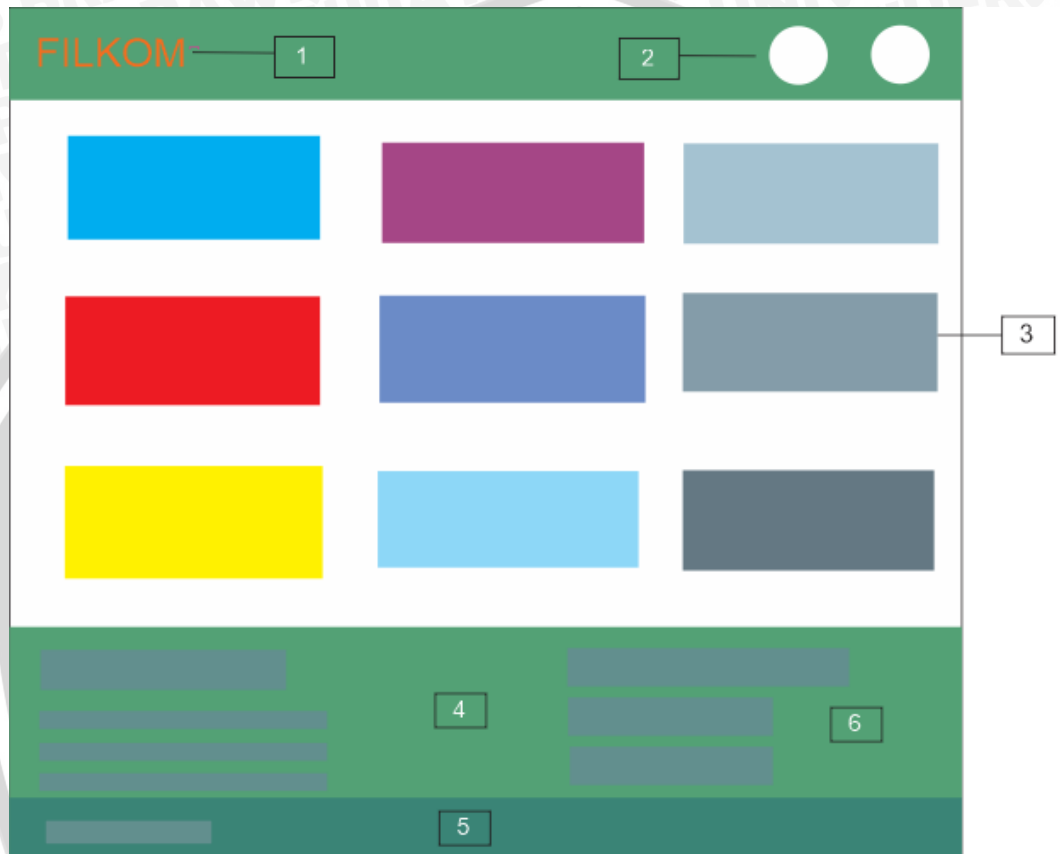
Gambar 4.14 Halaman Hasil Pencarian dan Kategori

Keterangan Gambar 4.14:

1. Logo FILKOM sebagai identitas sistem
2. Menu kategori dan aktivasi rekomendasi
3. Gambar barang
4. Nama barang
5. Harga barang
6. Pernyataan pembuatan aplikasi
7. Kerjasama *E-commerce*
8. Hak cipta

4.4.3 Halaman Kategori Barang

Halaman hasil pencarian barang yang akan dirancang memiliki 5 komponen penyusun halaman, yaitu logo FILKOM sebagai identitas, menu *top-navbar* sebagai navigasi untuk menuju halaman kategori, kategori barang, pernyataan dan sumber pada bagian *footer*. Visualisasi halaman dapat dilihat pada Gambar 4.15.



Gambar 4.15 Halaman Kategori Barang

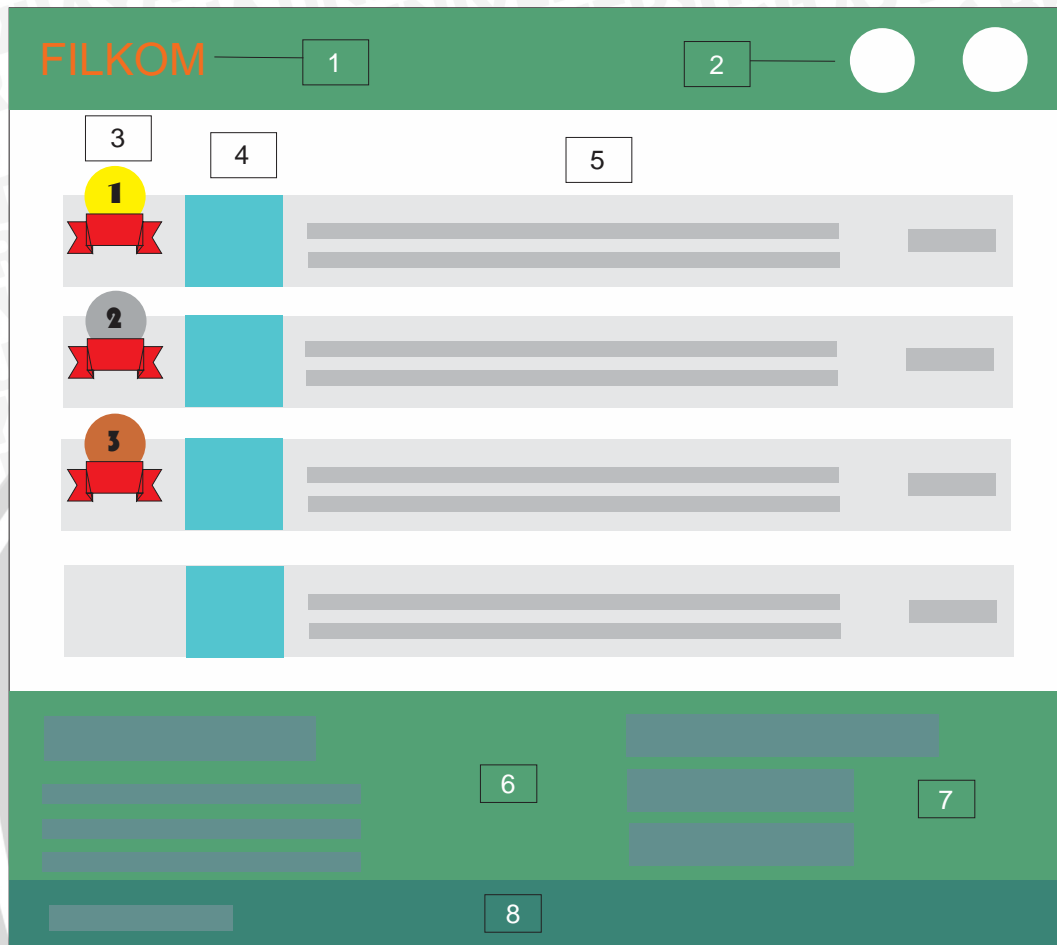
Keterangan Gambar 4.15:

1. Logo FILKOM sebagai identitas sistem
2. Menu kategori dan aktivasi rekomendasi
3. Kategori Barang
4. Pernyataan pembuatan aplikasi
5. Kerjasama *E-commerce*
6. Hak cipta

4.4.4 Halaman Hasil Rekomendasi

Halaman hasil pencarian barang yang akan dirancang memiliki 8 komponen penyusun halaman, yaitu logo FILKOM sebagai identitas, menu *top-navbar* sebagai navigasi untuk menuju halaman kategori, ranking barang, gambar barang, nama

barang, nilai preferensi barang, pernyataan dan sumber pada bagian *footer*. Visualisasi halaman dapat dilihat pada Gambar 4.16.



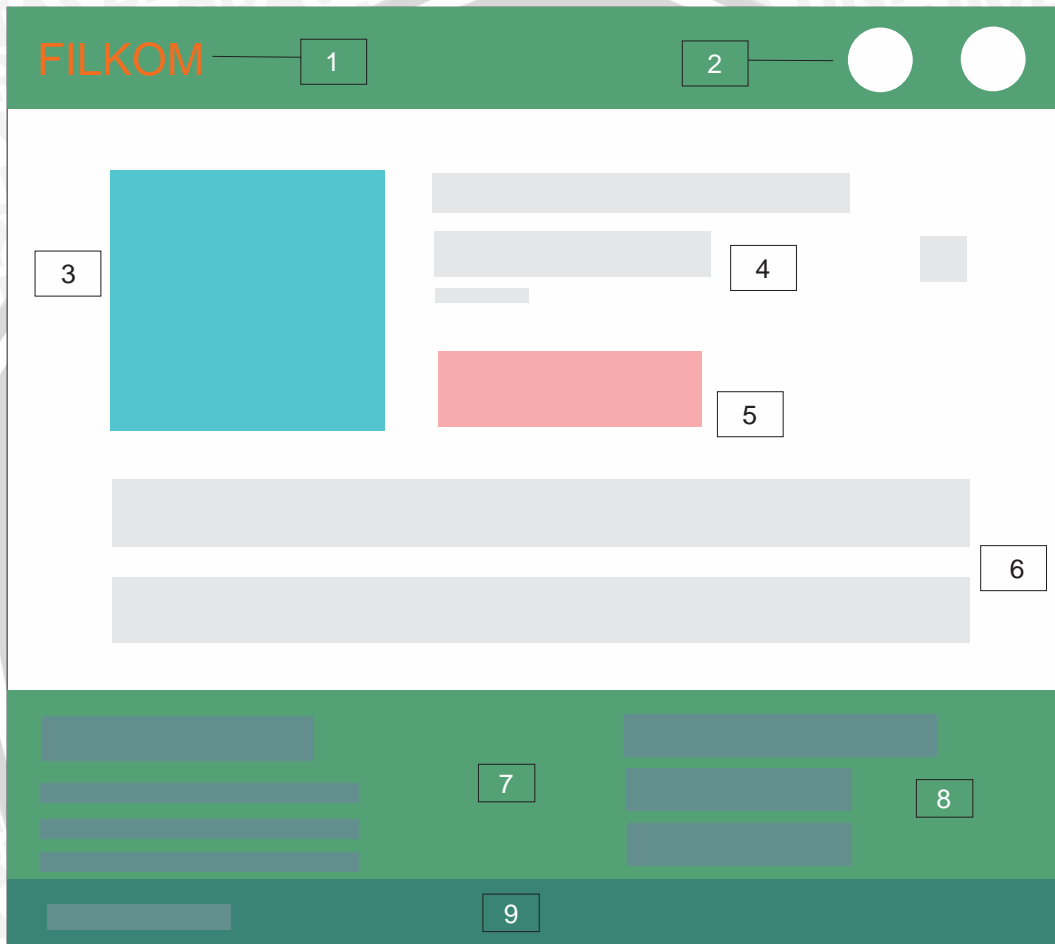
Gambar 4.16 Halaman Rekomendasi Barang

Keterangan Gambar 4.16:

1. Logo FILKOM sebagai identitas sistem
2. Menu kategori dan aktivasi rekomendasi
3. Gambar ranking barang
4. Gambar barang
5. Nama dan nilai preferensi barang
6. Pernyataan pembuatan aplikasi
7. Kerjasama *E-commerce*
8. Hak cipta

4.4.5 Halaman Detail Barang

Halaman hasil pencarian barang yang akan dirancang memiliki 9 komponen penyusun halaman, yaitu logo FILKOM sebagai identitas, menu *top-navbar* sebagai navigasi untuk menuju halaman kategori, gambar, nama, tombol ke sumber dan informasi barang, pernyataan dan sumber data pada bagian *footer*. Visualisasi halaman dapat dilihat pada Gambar 4.17.



Gambar 4.17 Halaman Detail Barang

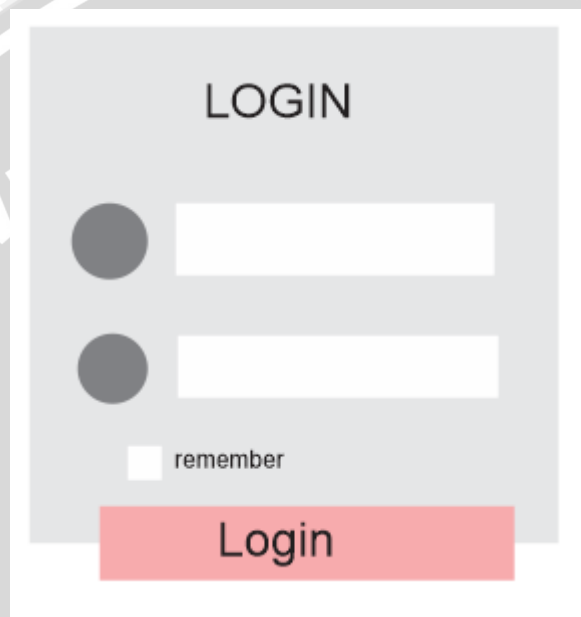
Keterangan Gambar 4.17:

1. Logo FILKOM sebagai identitas sistem
2. Menu kategori dan aktivasi rekomendasi
3. Gambar detail barang
4. Nama, sumber data, harga dan stok barang.
5. Tombol menuju sumber barang
6. Deskripsi lengkap barang

- 7. Pernyataan pembuatan aplikasi
- 8. Kerjasama *E-commerce*
- 9. Hak Cipta

4.4.6 Halaman Login Admin

Halaman *login* digunakan sebagai pintu masuk menuju halaman *admin*, jika terjadi kesalahan dalam pengisian *username* dan *password* maka pengguna akan diarahkan kembali ke halaman ini. Visualisasi halaman dapat dilihat pada Gambar 4.18.



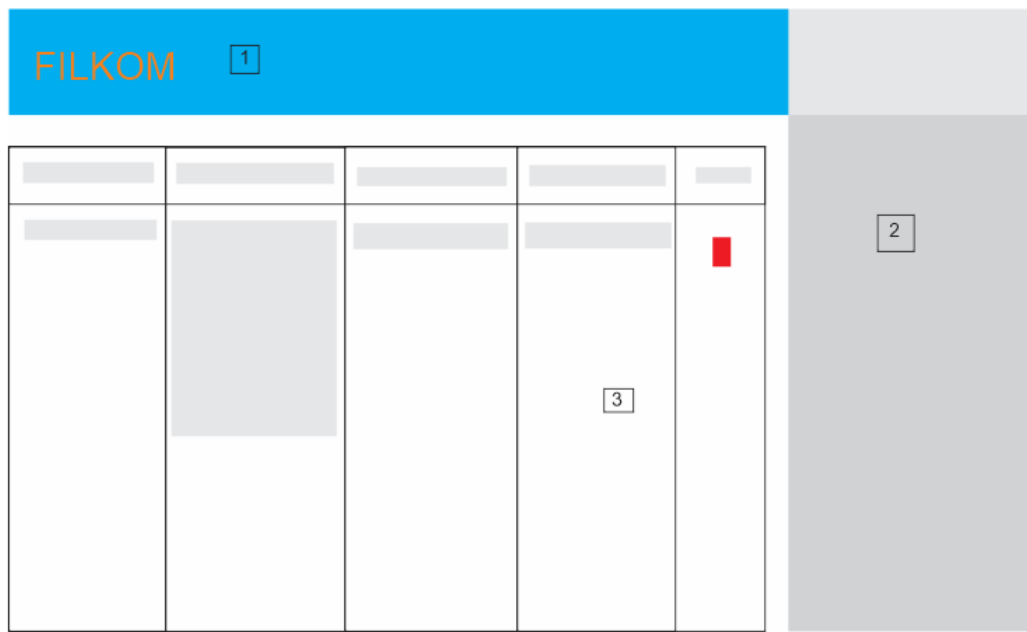
Gambar 4.18 Halaman Login Admin

Keterangan Gambar 4.18:

1. *Field Username*
2. *Field Password*
3. *Remember checkbox*
4. Tombol *Login*

4.4.7 Halaman Admin Kelola Data Barang

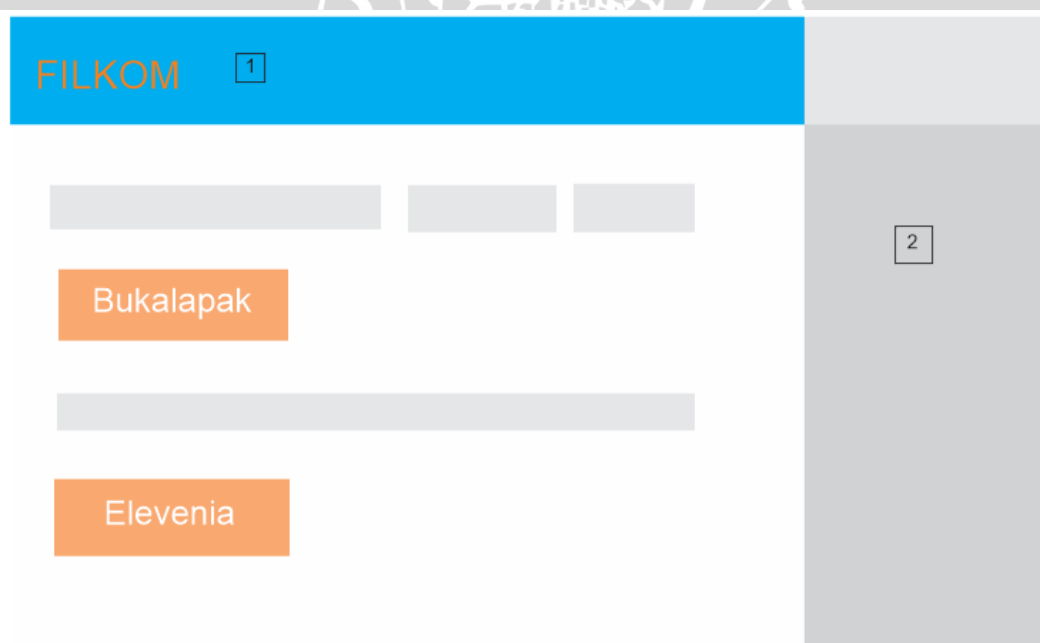
Halaman kelola barang hanya ditujukan untuk *admin* yang memiliki hak akses, adapun komponen penyusun halaman kelola data barang terdiri dari logo FILKOM, menu bagian kanan, dan data barang secara keseluruhan. Visualisasi halaman dapat dilihat pada Gambar 4.15.



Gambar 4.19 Halaman Kelola Barang

4.4.8 Halaman Admin *Scraping* Barang

Halaman *Scraping* hanya ditujukan untuk *admin* yang memiliki hak akses, adapun komponen penyusun halaman kelola data barang terdiri dari logo FILKOM, menu bagian kanan, dan *form* pengisian kriteria kata kunci yang akan diambil datanya. Visualisasi halaman dapat dilihat pada Gambar 4.15.



Gambar 4.20 Halaman *Scraping* Barang

BAB 5 IMPLEMENTASI

Bab ini membahas tahapan implementasi sistem Sistem Pendukung Keputusan Rekomendasi Barang *Online* Menggunakan Metode *Web scraping* dan *Fuzzy Simple Additive Weighting* (FSAW) sesuai dengan hasil yang telah diterapkan dari perancangan sistem yang telah dibuat. Pembahasan bab ini terdiri dari Batasan Implementasi, Implementasi *fuzzy simple additive weighting*, dan implementasi antarmuka.

5.1 Batasan Implementasi

Bagian ini menguraikan batasan-batasan penggunaan perangkat keras dan lunak yang dapat menjalankan sistem tanpa kendala, terdiri dari batasan perangkat keras (*hardware*), batasan perangkat lunak (*software*), dan batasan sumber daya manusia (*brainware*).

5.1.1 Batasan Perangkat Keras (Hardware)

Untuk menjalankan Sistem Pendukung Keputusan Rekomendasi Barang *Online* Menggunakan Metode *Web scraping* dan *Fuzzy Simple Additive Weighting* disarankan menggunakan sumber daya perangkat keras, dengan *spesifikasi* sebagai berikut:

1. *Processor*: Intel® Core™ i7-4700HQ CPU @ 2.40GHz
2. *RAM*: 4GB
3. *Harddisk*: Terpasang 1TB, minimum yang dirankan 40GB
4. *VGA*: Terpasang 4GB, minimum 512MB
5. *Keyboard* dan *mouse*
6. Resolusi layar monitor disarankan 1366×768 *pixels* atau lebih rendah

5.1.2 Batasan Perangkat Lunak (Software)

Untuk menjalankan Sistem Pendukung Keputusan Rekomendasi Barang *Online* Menggunakan Metode *Web scraping* dan *Fuzzy Simple Additive Weighting* disarankan menggunakan sumber daya perangkat lunak, dengan *spesifikasi* sebagai berikut:

1. *Operating System* Windows 10 Pro 64 bit
2. *XAMPP Server Version* 3.1.0
3. Aplikasi PHPStorm dari *Jetbrains Version* 8.0.1
4. *Database MySQL Version* 5.5.16
5. *Framework CodeIgniter Version* 3.1.0
6. Simple HTML DOM library *Version* 1.3.0
7. Materialize CSS library.

8. JQuery library Version 2.1.4

5.1.3 Batasan Sumber Daya Manusia (Brainware)

Untuk menjalankan Sistem Pendukung Keputusan Rekomendasi Barang *Online* Menggunakan Metode *Web scraping* dan *Fuzzy Simple Additive Weighting* disarankan menggunakan sumber daya manusia yang dapat menggunakan aplikasi *web* pada umumnya.

5.2 Implementasi *Fuzzy Simple Additive Weighting*

Bagian ini menjelaskan implementasi algoritma yang berkaitan dengan proses perhitungan *Simple Additive Weighting* pada Sistem Pendukung Keputusan Rekomendasi Barang *Online* Menggunakan Metode *Web Scraping* dan *Fuzzy Simple Additive Weighting*, perhitungan dipresentasikan dalam bentuk Kode Sumber dengan Bahasa pemrograman PHP dan *framework* CodeIgniter. Penjelasan terdiri dari implementasi algoritma *fuzzy*, implementasi algoritma normalisasi terbobot, implementasi algoritma preferensi.

5.2.1 Implementasi Algoritma *Fuzzy*

Matriks keputusan *fuzzy* dibuat berdasarkan data barang yang tersimpan didalam *database* dan belum dilakukan proses *fuzzification*, lalu data yang telah diubah, di masukkan kedalam tabel kriteria. Tabel kriteria terdiri c1, c2, c3, c4, c5, c6 dan c7 yang masing-masing mengindikasikan kriteria barang. Kode Sumber 5.1 adalah hasil implementasi perhitungan dari algoritma *fuzzy*.

```

1. $kriteria = $this->db->get('sk_kriteria');
2. $a = $this->db->where('id >=', $kriteria->last_row()-
   >id_barang)->get('sk_barang');
3. foreach ($a->result() as $b) {
4. //           penentuan nilai bobot harga
5.     if ($b->harga >= 0 && $b->harga <= 50000) {
6.         $c1 = 0.04;
7.     } elseif ($b->harga > 50000 && $b->harga <=
   500000) {
8.         $c1 = 0.09;
9.     } elseif ($b->harga > 500000 && $b->harga <=
   5000000) {
10.        $c1 = 0.15667;
11.    } elseif ($b->harga > 5000000 && $b->harga <=
   10000000) {
12.        $c1 = 0.25667;
13.    } else {
14.        $c1 = 0.456667;

```

```
15.     }
16.     //           penentuan nilai bobot berat
17.     if ($b->berat >= 0 && $b->berat <= 100) {
18.         $c2 = 0.111111;
19.     } elseif ($b->berat > 100 && $b->berat <= 2000) {
20.         $c2 = 0.277778;
21.     } else {
22.         $c2 = 0.611111;
23.     }
24.     //           penentuan nilai bobot stok
25.     if ($b->stok >= 0 && $b->stok <= 5) {
26.         $c3 = 0.04;
27.     } elseif ($b->stok > 5 && $b->stok <= 20) {
28.         $c3 = 0.09;
29.     } elseif ($b->stok > 20 && $b->stok <= 100) {
30.         $c3 = 0.15667;
31.     } elseif ($b->stok > 100 && $b->stok <= 500) {
32.         $c3 = 0.25667;
33.     } else {
34.         $c3 = 0.456667;
35.     }
36.     //           penentuan nilai bobot lokasi
37.     if (strpos($b->lokasi, "Jakarta") == TRUE) {
38.         $c4 = 0.111111;
39.     } elseif (strpos($b->lokasi, "Jawa") == TRUE |
40. strpos($b->lokasi, "Yogyakarta") == TRUE) {
41.         $c4 = 0.277778;
42.     } else {
43.         $c4 = 0.611111;
44.     }
45.     //           penentuan nilai bobot deskripsi
46.     if (strlen($b->deskripsi) >= 0 && strlen($b-
47. >deskripsi) <= 50) {
48.         $c5 = 0.0625;
49.     } elseif (strlen($b->deskripsi) > 50 &&
50. strlen($b->deskripsi) <= 200) {
51.         $c5 = 0.145833;
```

```
49.         } elseif (strlen($b->deskripsi) > 200 &&
    strlen($b->deskripsi) <= 1000) {
50.             $c5 = 0.270833;
51.         } else {
52.             $c5 = 0.520833;
53.         }
54.     //         penentuan nilai bobot rating
55.     if ($b->rating >= 0 && $b->rating <= 20) {
56.         $c6 = 0.04;
57.     } elseif ($b->rating > 20 && $b->rating <= 40) {
58.         $c6 = 0.09;
59.     } elseif ($b->rating > 40 && $b->rating <= 60) {
60.         $c6 = 0.15667;
61.     } elseif ($b->rating > 60 && $b->rating <= 80) {
62.         $c6 = 0.25667;
63.     } else {
64.         $c6 = 0.456667;
65.     }
66.     //         penentuan nilai bobot terjual
67.     if ($b->terjual >= 0 && $b->terjual <= 2) {
68.         $c7 = 0.04;
69.     } elseif ($b->terjual > 2 && $b->terjual <= 10) {
70.         $c7 = 0.09;
71.     } elseif ($b->terjual > 10 && $b->terjual <= 100) {
72.         $c7 = 0.15667;
73.     } elseif ($b->terjual > 100 && $b->terjual <= 500) {
74.         $c7 = 0.25667;
75.     } else {
76.         $c7 = 0.456667;
77.     }
78.     $data = array(
79.         'id_barang' => $b->id,
80.         'c1' => $c1,
81.         'c2' => $c2,
82.         'c3' => $c3,
83.         'c4' => $c4,
```

```

83.         'c5' => $c5,
84.         'c6' => $c6,
85.         'c7' => $c7
86.     );
87.     $q = $this->db->where('id_barang', $b->id)-
>get('sk_kriteria')->row_array();
88.     if ($q['id_barang'] == $b->id) {
89.         $this->db->where('id_barang', $b->id)-
>update('sk_kriteria', $data);
90.     } else {
91.         $this->db->insert('sk_kriteria', $data);
92.     }
93.     echo $b->id . "-" . $c1 . "--" . $c2 . "---" .
$c3 . "---" . $c4 . "---" . $c5 . "---" . $c6 . "---" . $c7
. "<br>";
94.     }
95.     echo "Berhasil mem-fuzzifikasi <a
href='".site_url('admin/dashboard')."'">Kembali</a>";
96.     }

```

Kode Sumber 5.1 Implementasi Fuzzy

Penjelasan implementasi matriks *fuzzy* pada Kode Sumber 5.1 adalah sebagai berikut:

1. Baris 1 merupakan fungsi untuk memanggil tabel kriteria
2. Baris 2 merupakan fungsi untuk memanggil tabel barang dengan kondisi, proses matriks *fuzzy* dilakukan pada ID barang yang lebih dari ID terakhir pada tabel kriteria.
3. Baris 3 merupakan perulangan untuk memanggil data barang.
4. Baris 4 – 15 merupakan pengondisian untuk mengubah data harga barang kedalam bilangan *fuzzy*, dan hasil disimpan ke dalam variabel c1.
5. Baris 16 – 23 merupakan pengondisian untuk mengubah data berat barang kedalam bilangan *fuzzy*, dan hasil disimpan ke dalam variabel c2.
6. Baris 24 – 35 merupakan pengondisian untuk mengubah data stok barang kedalam bilangan *fuzzy*, dan hasil disimpan ke dalam variabel c3.
7. Baris 36 – 43 merupakan pengondisian untuk mengubah data lokasi barang kedalam bilangan *fuzzy*, dan hasil disimpan ke dalam variabel c4.
8. Baris 44 – 53 merupakan pengondisian untuk mengubah data deskripsi barang kedalam bilangan *fuzzy*, dan hasil disimpan ke dalam variabel c5.

9. Baris 54 – 65 merupakan pengondisian untuk mengubah data *rating* barang kedalam bilangan *fuzzy*, dan hasil disimpan ke dalam variabel c6.
10. Baris 66 – 77 merupakan pengondisian untuk mengubah data terjual barang kedalam bilangan *fuzzy*, dan hasil disimpan ke dalam variabel c7.
11. Baris 78 – 87 merupakan proses penyimpanan seluruh variabel dari c1 hingga c7 kedalam *array* data.
12. Baris 88 – 97 merupakan proses *update* jika data kriteria sudah ada, dan *insert* jika data kriteria belum ada, serta keterangan data *fuzzy* berhasil di ubah.

5.2.2 Implementasi Algoritma Normalisasi

Matriks normalisasi terbobot di hitung dengan cara mencari nilai maksimum kriteria dibagi dengan nilai kriteria atau nilai kriteria dibagi dengan minimum setiap kriteria. Implementasi dari matriks normalisasi terbobot pada bagian *controller* dapat dilihat pada Kode Sumber 5.2, dan bagian *view* dapat dilihat pada Kode Sumber 5.3.

```

1. $keyword = $_GET['q'];
2. $max      =      $this->db->select_max('c3','maxC3')-
>select_max('c5','maxC5')->select_max('c6','maxC6')-
>select_max('c7','maxC7')-
>like('sk_barang.nama',$keyword,'after')-
>where('sk_barang.terjual >',0)-
>join('sk_barang','sk_barang.id =
sk_kriteria.id_barang')->limit(50)->get('sk_kriteria');
3.          $min    =      $this->db->select_min('c1','minC1')-
>select_min('c2','minC2')->select_min('c4','minC4')-
>like('sk_barang.nama',$keyword,'after')-
>where('sk_barang.terjual >',0)-
>join('sk_barang','sk_barang.id =
sk_kriteria.id_barang')->limit(50)->get('sk_kriteria');
4.          $this->data['max'] = $max->row_array();
5.          $this->data['min'] = $min->row_array();
6.          $this->data['kriteria'] =      $this->db-
>like('sk_barang.nama',$keyword,'after')-
>where('sk_barang.terjual >',0)-
>join('sk_barang','sk_barang.id =
sk_kriteria.id_barang')->limit(50)->get('sk_kriteria');
7.          $this->tampil('halaman/public/normalisasi');

```

Kode Sumber 5.2 Implementasi *Controller* Algoritma Normalisasi

Penjelasan implementasi algoritma normalisasi pada Kode Sumber 5.2 adalah sebagai berikut:

1. Baris 1 merupakan proses mendapatkan nilai dari *URI*

2. Baris 2 merupakan proses untuk mendapatkan nilai maksimum masing-masing kriteria dari hasil penggabungan tabel barang dan kriteria.
3. Baris 3 merupakan proses untuk mendapatkan nilai minimum masing-masing kriteria dari hasil penggabungan tabel barang dan kriteria.
4. Baris 4-5 mendefinisikan variabel max dan min untuk dipanggil kembali pada halaman *view*
5. Baris 6 merupakan proses untuk mendapatkan data kriteria dan barang dari hasil penggabungan tabel barang dan tabel kriteria.
6. Baris 7 adalah fungsi untuk memanggil halaman *view* normalisasi, penjelasan halaman *view* normalisasi dapat dilihat pada Kode Sumber 5.3.

```

1. <tbody>
2.     <?php
3.         $no = 0;
4.         foreach($kriteria->result_array() as $dt2) {
5.             echo "<tr>
6.                 <td>". $dt2['id_barang']. "</td>
7.                 <td>". $this->db->where('id', $dt2['id_barang'])->get('sk_barang')->row()->nama. "</td>
8.
9.                 <td>". round($min['minC1']/$dt2['c1'], 6). "</td>
10.                <td>". round($min['minC2']/$dt2['c2'], 6). "</td>
11.                <td>". round($dt2['c3']/$max['maxC3'], 6). "</td>
12.                <td>". round($min['minC4']/$dt2['c4'], 6). "</td>
13.                <td>". round($dt2['c5']/$max['maxC5'], 6). "</td>
14.                <td>". round($dt2['c6']/$max['maxC6'], 6). "</td>
15.                <td>". round($dt2['c7']/$max['maxC7'], 6). "</td>
16.            </tr>";
17.         }
18.     </tbody>

```

Kode Sumber 5.3 Implementasi View Algoritma Normalisasi

Penjelasan implementasi algoritma normalisasi pada Kode Sumber 5.3 adalah sebagai berikut:

1. Baris 4 merupakan proses perulangan untuk mendapatkan data dari tabel kriteria
2. Baris 6 menampilkan id barang dari tabel kriteria.
3. Baris 7 merupakan menampilkan nama barang dari tabel barang, berdasarkan id barang pada tabel kriteria.
4. Baris 8,9 dan 11 mendefinisikan variabel *max* dan *min* untuk dipanggil kembali pada halaman *view*
5. Baris 10, 12-14 merupakan proses untuk mendapatkan data barang dari hasil penggabungan tabel barang dan tabel kriteria.

5.2.3 Implementasi Algoritma Preferensi

Algoritma preferensi di hitung dengan mengalikan nilai masing-masing data pada matriks bobot ternormalisasi dengan nilai bobot yang telah ditentukan. Implementasi dari nilai preferensi pada bagian controller dapat dilihat pada Kode Sumber 5.4, dan bagian *view* dapat dilihat pada Kode Sumber 5.5.

```

1. $this->data['b1'] = 0.25;
2. $this->data['b2'] = 0.07;
3. $this->data['b3'] = 0.07;
4. $this->data['b4'] = 0.08;
5. $this->data['b5'] = 0.13;
6. $this->data['b6'] = 0.15;
7. $this->data['b7'] = 0.25;
8. $keyword = $_GET['q'];
9. $max      =      $this->db->select_max('c3','maxC3')-
>select_max('c5','maxC5')->select_max('c6','maxC6')-
>select_max('c7','maxC7')-
>like('sk_barang.nama',$keyword,'after')-
>where('sk_barang.terjual >',0)-
>join('sk_barang','sk_barang.id      =
sk_kriteria.id_barang')->limit(50)->get('sk_kriteria');
10. $min      =      $this->db->select_min('c1','minC1')-
>select_min('c2','minC2')->select_min('c4','minC4')-
>like('sk_barang.nama',$keyword,'after')-
>where('sk_barang.terjual >',0)-
>join('sk_barang','sk_barang.id      =
sk_kriteria.id_barang')->limit(50)->get('sk_kriteria');
11. $this->data['max'] = $max->row_array();
12. $this->data['min'] = $min->row_array();

```



```

13. $this->data['ranking'] = $this->db->
    >like('sk_barang.nama',$keyword,'after')->
    >where('sk_barang.terjual >',0)->
    >join('sk_barang','sk_barang.id =
    sk_kriteria.id_barang')->limit(50)->get('sk_kriteria');
14. $this->tampil('halaman/public/preferensi');

```

Kode Sumber 5.4 Implementasi *Controller* Algoritma Preferensi

Penjelasan implementasi *controller* algoritma preferensi pada Kode Sumber 5.4 adalah sebagai berikut:

1. Baris 1 - 7 merupakan nilai bobot kriteria, yang disimpan dalam variabel *b1* hingga *b7*.
2. Baris 9 merupakan proses untuk mendapatkan nilai maksimum masing-masing kriteria dari hasil penggabungan tabel barang dan kriteria.
3. Baris 10 merupakan proses untuk mendapatkan nilai minimum masing-masing kriteria dari hasil penggabungan tabel barang dan kriteria.
4. Baris 11 - 12 mendefinisikan variabel *max* dan *min* untuk dipanggil kembali pada halaman *view*
5. Baris 13 merupakan proses untuk mendapatkan data kriteria dan barang dari hasil penggabungan tabel barang dan tabel kriteria.
6. Baris 14 adalah fungsi untuk memanggil halaman *view* normalisasi, penjelasan halaman *view* normalisasi dapat dilihat pada Kode Sumber 5.5.

```

1. <tbody>
2.   <?php
3.     $no = 1;
4.     foreach ($ranking->result_array() as $dt3) {
5.       $info = $this->db->where('id',$dt3['id_barang'])->
        >get('sk_barang');
6.       $pref = round((( $min['minC1']/$dt3['c1'] ) * $b1) +
7.                   (( $min['minC2']/$dt3['c2'] ) * $b2) +
8.                   (( $dt3['c3']/$max['maxC3'] ) * $b3) +
9.                   (( $min['minC4']/$dt3['c4'] ) * $b4) +
10.                  (( $dt3['c5']/$max['maxC5'] ) * $b5) +
11.                  (( $dt3['c6']/$max['maxC6'] ) * $b6) +
12.                  (( $dt3['c7']/$max['maxC7'] ) * $b7)
13.       , 6);
14.       echo " <tr>

```

```

15.         <td><a
      href="'.site_url('halaman/details/' . $info->row()-
      >id)."'></a></td>
16.         <td>" . $info->row()->nama . "</td>
17.         <td>" . $pref . "</td>
18.     </tr> ";
19.     $no++;
20.     }
21.     ?>
22. </tbody>
23. <script>
24.     $(document).ready(function()
25.     {
26.         $("#ranking").tablesorter(      {sortList:
      [[2,1],[1,1]] } );
27.     }
28.     );
29. </script>

```

Kode Sumber 5.5 Implementasi *View* Algoritma Preferensi

Penjelasan Implementasi Algoritma Normalisasi pada Kode Sumber 5.3 adalah sebagai berikut:

1. Baris 4 merupakan proses perulangan untuk mendapatkan data dari tabel kriteria
2. Baris 5 mendefinisikan id barang dari tabel kriteria kedalam variabel info.
3. Baris 6 - 13 adalah proses perhitungan nilai preferensi dan didefinisikan kedalam variabel pref.
4. Baris 15 - 17 menampilkan data barang, berupa gambar, nama, dan nilai preferensinya.
5. Baris 23 - 29 merupakan proses untuk mengurutkan data barang berdasarkan nilai preferensinya, menggunakan *library JQuery tablesorter*.

5.3 Implementasi Proses *Scraping* Data

Proses *scraping* data yang diterapkan pada Sistem Pendukung Keputusan Rekomendasi Barang *Online* Menggunakan Metode *Web scraping* dan *Fuzzy Simple Additive Weighting* ini menggunakan library Simple HTML DOM, yang otomatis melacak struktur pohon HTML, untuk selanjutnya data dapat diolah sesuai keinginan. Penjelasan sub-bab ini terdiri dari implementasi *scraping* situs bukalapak.com dan, implementasi *scraping* situs elevenia.com.

5.3.1 Scraping Situs Bukalapak

Data didapatkan dari situs *E-commerce* Bukalapak.com, untuk lebih spesifik, data yang akan diolah didapatkan dari hasil pencarian situs bukalapak. Setiap barang yang dimasukkan kedalam *database*, berdasarkan kata kunci dan nomor halaman, data yang dapat dimasukkan oleh administrator mulai dari 1 hingga n halaman. Implementasi dari *scraping* situs bukalapak dapat dilihat pada Kode Sumber 5.6.

```

1. public function bukalapak(){
2.   if (!$this->session->userdata('username')) {
3.     $this->session->set_flashdata('error','Login Dulu');
4.     redirect('admin/user/login');
5.   } else {
6.     $mulai = $_POST['start'];
7.     $selesai = $_POST['end'];
8.     $keyword = $_POST['keyword'];
9.     for ($i = $mulai; $i <= $selesai; $i++) {
10.      $url =
11.      "https://www.bukalapak.com/products?page=".$i."&search%
12.      5Bkeywords%5D=".$keyword."&sizes%5Bgeneral%5D=83677&utf
13.      8=%E2%9C%93";
14.      $html = file_get_html($url);
15.      $ret = $html->find('li.product');
16.      echo "Scrape halaman ke - ". $i . "</br>";
17.      foreach ($ret as $tag) {
18.        //dapatkan nama barang
19.        $nama = $tag->find('div.product-media a', 0)-
20.        >title;
21.        echo $nama."<i style='color:red'>
22.        Ditambahkan</i>";
23.        //dapatkan link barang
24.        $link = $tag->find('div.product-media a',
25.        0)->href;
26.        //dapatkan harga barang
27.        $harga = $tag->find('div.product-price
28.        span.amount', 0)->plaintext;
29.        //link gambar barang
30.        $gambar = $tag->find('.product-media img',
31.        0)->src;

```

```
24.         $lokasi = $tag->find('div.user-city', 0)-
>plaintext;
25.         $feedback = $tag->find('a.user-feedback-
summary', 0)->plaintext;
26.         $detailurl =
file_get_html('https://www.bukalapak.com' . $link);
27.         $det = $detailurl->find('div[id=display_product_desktop]');
28.         foreach ($det as $detail) {
29.             $deskripsi = $detail->find('.js-
collapsible', 0)->plaintext;
30.             $stok = $detail->find('.product-stock',
0)->plaintext;
31.             $berat = $detail->find('.product-spec',
0)->children(5)->plaintext;
32.         }
33.         $trim = explode('/', trim($link));
34.         $kategori = $trim[2];
35.         $arr = explode(" ", $feedback, 2);
36.         $data = array(
37.             'nama' => $nama,
38.             'harga' => filter_var($harga,
FILTER_SANITIZE_NUMBER_INT),
39.             'kategori' => $kategori,
40.             'lokasi' => $lokasi,
41.             'berat' => filter_var($berat,
FILTER_SANITIZE_NUMBER_INT),
42.             'rating' => $arr[0],
43.             'terjual' => "",
44.             'deskripsi' => $deskripsi,
45.             'stok' => filter_var($stok,
FILTER_SANITIZE_NUMBER_INT),
46.             'gambar' => $gambar,
47.             'sumber' => "https://www.bukalapak.com"
. $link
48.         );
49.         $query = $this->db->where('nama',
$nama)->get('sk_barang')->row_array();
50.         if ($query["nama"] == $nama) {
51.             echo "sudah ada" . "</br>";
```

```

52.         $this->db->where('nama', $nama);
53.         $this->db->update('sk_barang', $data);
54.         } else {
55.         $this->db->insert('sk_barang', $data);
56.         }
57.     }
58. }
59.         echo 'Berhasil menambahkan barang <a
href="'.site_url('admin/scrape/bukalapak_info').'">Upda
te Info</a>';
60.     }
61. }

```

Kode Sumber 5.6 Implementasi *Scraping* Bukalapak

Penjelasan implementasi *scraping* bukalapak pada Kode Sumber 5.6 adalah sebagai berikut:

1. Baris 2-5 merupakan validasi *admin*, jika session tidak ditemukan maka pengguna akan diarahkan ke halaman login.
2. Baris 6-8 definisi dari variabel *keyword*, *start* dan *end* yang didapatkan dari inputan *admin* pada halaman *scrape*.
3. Baris 9 merupakan perulangan untuk mendapatkan data hingga halaman ke \$selesai.
4. Baris 10 merupakan alamat *URI* pencarian situs bukalapak.
5. Baris 14 – 35 adalah proses mendapatkan nilai kriteria barang pada situs bukalapak, dan di simpan dalam bentuk variabel sesuai kriteria
6. Baris 36 – 48 merupakan penyimpanan variabel-variabel yang didapatkan kedalam bentuk array, supaya lebih mudah untuk dimasukkan kedalam *database*.
7. Baris 49 – 58 merupakan proses *update* jika barang telah ada didalam *database* dan *insert* jika barang belum ada.
8. Baris 59 Pesan berhasil menambahkan atau mengubah data barang, dilanjutkan link untuk menambahkan info detail barang, karna info detail barang tidak berada pada halaman yang sama pada situs bukalapak.

5.3.2 *Scraping* Situs Elevenia

Data barang didapatkan dari situs *E-commerce* Elevenia.com. Untuk lebih spesifik, data yang akan diolah didapatkan dari setiap kata kunci pencarian. Setiap barang yang dimasukkan kedalam *database*, berdasarkan nama kategori yang telah dimasukkan *admin*. Implementasi dari *scraping* situs elevenia dapat dilihat pada Kode Sumber 5.7.

```
1. public function elevenia() {
2.     if (!$this->session->userdata('username')) {
3.         $this->session->set_flashdata('error', 'Login Dulu');
4.         redirect('admin/user/login');
5.     } else {
6.         $keyword = $_POST['keyword'];
7.         $url = "http://elevenia.com/search?q=".$keyword;
8.         $kategori = $_POST['kategori'];
9.         $html = file_get_html($url);
10.        $ret = $html->find('ul.albumList li');
11.        foreach ($ret as $tag) {
12.            $link = $tag->find('a', 0)->href;
13.            $nama = $tag->find('a img', 0)->alt;
14.            echo          $nama."<i style='color:red'>
                Ditambahkan</i>";
15.            $gambar = $tag->find('a img', 0)->src;
16.            $harga = $tag->find('.price strong.nottranslate',
                0)->plaintext;
17.            $toko = $tag->find('.stroename a', 0)->plaintext;
18.            $terjual = $tag->find('div.rankingArea a', 0)-
                >plaintext;
19.            $html = file_get_html($link);
20.            $det = $html->find('.productContent');
21.            foreach ($det as $detail) {
22.                $rating = $detail->find('.sellerInfo', 0)-
                >children(2)->plaintext;
23.                $data = array(
24.                    'nama' => $nama,
25.                    'harga' => filter_var($harga,
                FILTER_SANITIZE_NUMBER_INT),
26.                    'kategori' => $kategori,
27.                    'lokasi' => 'Yogyakarta',
28.                    'berat' => rand(100, 500),
29.                    'rating' => filter_var($rating,
                FILTER_SANITIZE_NUMBER_INT),
30.                    'terjual' => filter_var($terjual,
                FILTER_SANITIZE_NUMBER_INT),
```

```

31.         'deskripsi' => "",
32.         'stok' => rand(1, 20),
33.         'gambar' => $gambar,
34.         'sumber' => $link
35.     );
36.     $query = $this->db->where('nama', $nama)-
    >get('sk_barang')->row_array();
37.     if ($query["nama"] == $nama) {
38.         echo "sudah ada" . "</br>";
39.         $this->db->where('nama', $nama);
40.         $this->db->update('sk_barang', $data);
41.     } else {
42.         $this->db->insert('sk_barang', $data);
43.     }
44. }
45. }
46. }
47.     echo "Berhasil menambahkan barang <a
    href='".site_url('admin/dashboard')."'">Kembali</a>";
48. }

```

Kode Sumber 5.7 Implementasi *Scraping* Elevenia

Penjelasan implementasi *scraping* bukalapak pada Kode Sumber 5.7 adalah sebagai berikut:

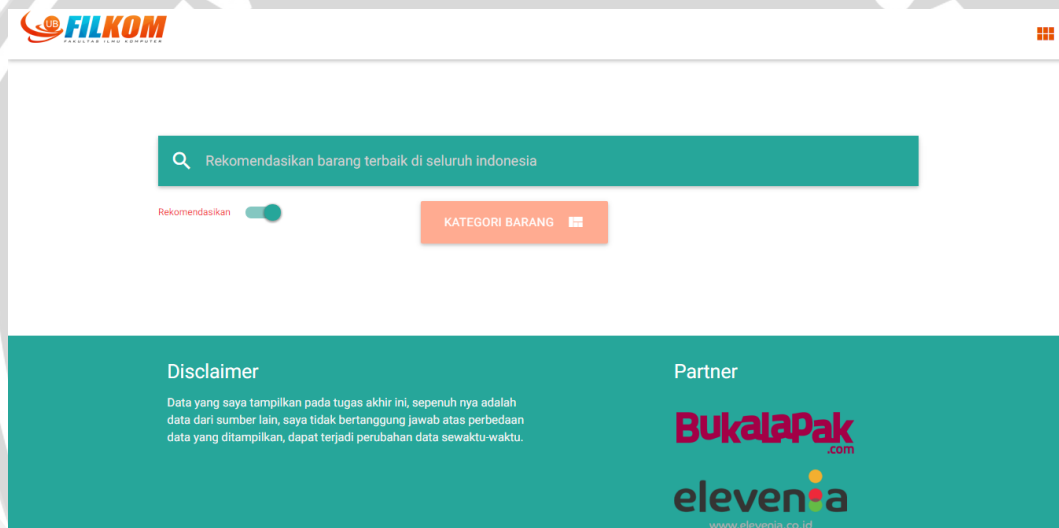
1. Baris 2-5 merupakan validasi *admin*, jika *session* tidak ditemukan maka pengguna akan diarahkan ke halaman *login*.
2. Baris 6-9 definisi dari variabel *keyword* yang berisi kata kunci pencarian, yang didapatkan dari inputan *admin* pada halaman *scrape*.
3. Baris 10 merupakan perulangan untuk mendapatkan data hingga halaman selesai diambil datanya.
4. Baris 11 – 21 adalah proses mendapatkan nilai kriteria barang pada situs elevenia, dan di simpan dalam bentuk variabel sesuai kriteria.
5. Baris 22 - 34 merupakan penyimpanan variabel-variabel yang didapatkan kedalam bentuk *array*, supaya lebih mudah untuk dimasukkan kedalam *database*.
6. Baris 35 – 45 merupakan proses *update* jika barang telah ada didalam *database* dan *insert* jika barang belum ada.
7. Baris 46 Pesan berhasil menambahkan atau mengubah data barang.

5.4 Implementasi Antarmuka

Antarmuka yang telah dibuat pada bab 4 selanjutnya akan divisualisasikan kedalam halaman web, antarmuka halaman menggunakan *library materializecss* untuk mendapatkan desain material yang digunakan oleh *google*. Implementasi antarmuka untuk *user non-admin* adalah implementasi halaman utama pencarian, halaman barang, halaman detail barang, halaman perangkan dan halaman kategori. Sedangkan implementasi antarmuka *admin* terdiri dari halaman *login*, halaman *scraping* barang, dan halaman kelola barang.

5.4.1 Antarmuka Halaman Utama Pencarian

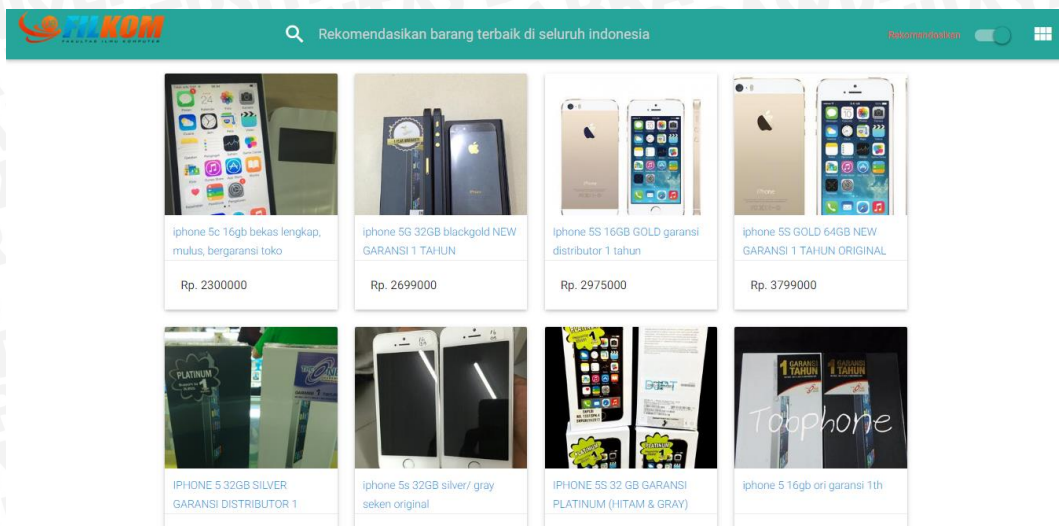
Halaman utama merupakan halaman yang pertama kali diakses oleh *admin* ataupun *non-admin*. Pada halaman ini terdapat kolom pencarian yang mendominasi halaman, hal ini bertujuan untuk memudahkan pengguna untuk memasukkan kata kunci pencarian, lalu ada tombol *switch* untuk rekomendasi. Tampilan halaman utama pencarian dapat dilihat pada Gambar 5.1.



Gambar 5.1 Antarmuka Halaman Utama Pencarian

5.4.2 Antarmuka Halaman Barang

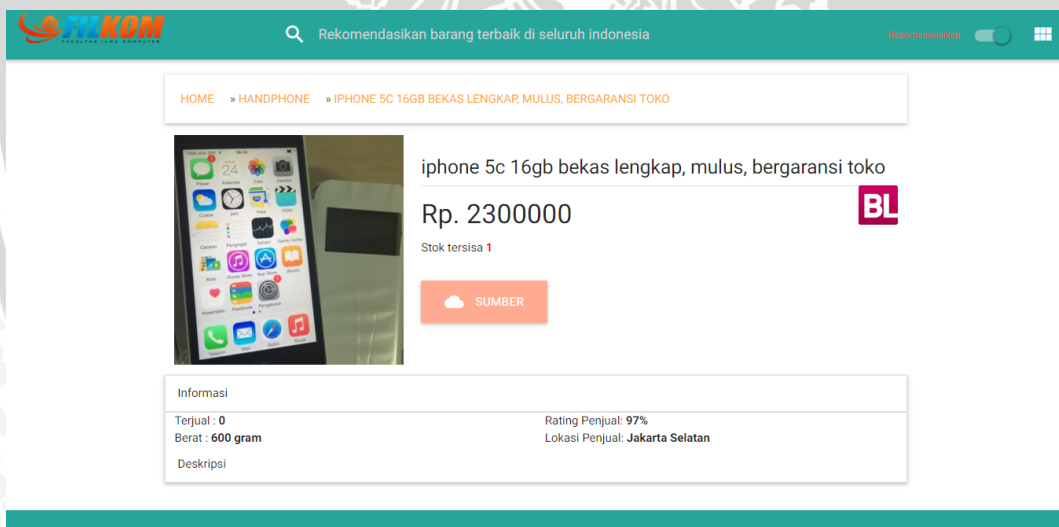
Halaman barang merupakan hasil pencarian dari halaman utama yang jika tombol *switch* tidak diaktifkan. Halaman barang berisi informasi singkat tentang barang dicari, yaitu berupa informasi gambar, nama dan harga barang. Tampilan halaman barang dapat dilihat pada Gambar 5.2.



Gambar 5.2 Antarmuka Halaman Barang

5.4.3 Antarmuka Halaman Detail Barang

Halaman detil barang merupakan halaman yang berisi informasi lengkap dari data barang, mulai dari gambar, harga, nama, kategori, sumber, stok, dan informasi kriteria lainnya. Untuk melihat tampilan dari halaman detil barang lihat Gambar 5.3.









Gambar 5.3 Antarmuka Halaman Detail Barang

5.4.4 Antarmuka Halaman Perankingan

Halaman perankingan adalah halaman yang digunakan untuk melihat ranking barang berdasarkan perhitungan *Fuzzy Simple Additive Weighting*, halaman akan aktif jika pengguna menekan tombol *switch* ke mode *on*. Untuk melihat tampilan dari halaman detil barang, lihat Gambar 5.4.

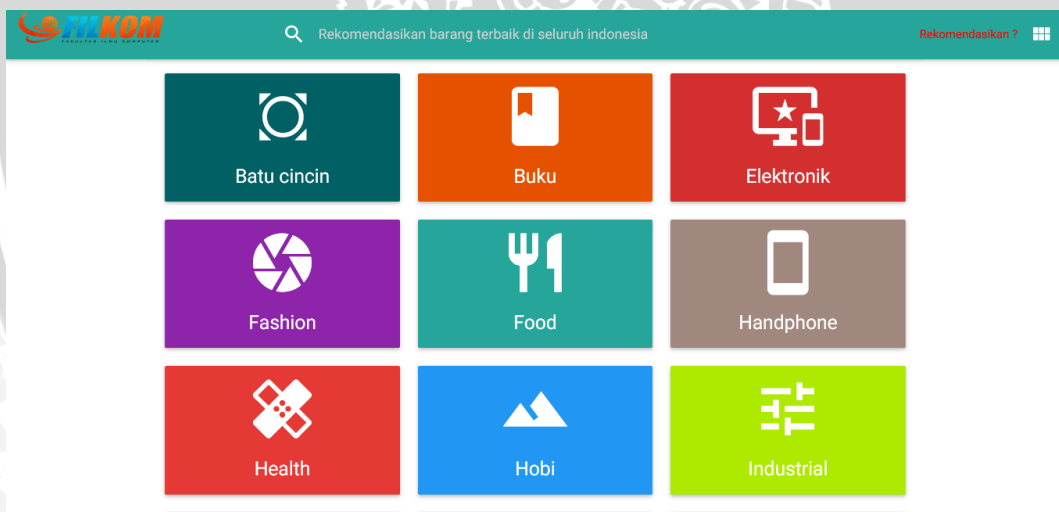


Gambar	Nama	Ranking
	 IPHONE 5 32GB SILVER - GARANSI 1 TAHUN DISTRIBUTOR	0.958
	 IPHONE 5 64GB SILVER - GARANSI 1 TAHUN DISTRIBUTOR	0.928212
	 iPhone 5s 64gb space gray / grey	0.860599

Gambar 5.4 Antarmuka Halaman Perankingan

5.4.5 Antarmuka Halaman Kategori

Halaman kategori merupakan halaman yang digunakan untuk menampilkan semua kategori barang, untuk mempermudah pengguna dalam memilih barang, sesuai dengan kategori yang disukai. Tampilan dari halaman kategori dapat dilihat pada Gambar 5.5.



Gambar 5.5 Antarmuka Halaman Kategori

5.4.6 Antarmuka Halaman Login Admin

Halaman login *admin* adalah halaman yang khusus digunakan bagi *admin* yang memiliki *username* dan *password*, halaman ini digunakan untuk memasuki sistem administrasi data, seperti proses *scraping*, kelola data barang, dan *fuzzifikasi* data. Untuk melihat tampilan dari halaman *login*, lihat pada Gambar 5.6.

Login

Login Dulu

Remember Me

LOG IN

Gambar 5.6 Antarmuka Halaman Login

5.4.7 Antarmuka Halaman *Scraping* Data Barang

Halaman scraping data barang merupakan halaman yang digunakan *admin* untuk mendapatkan data barang dari situs bukalapak dan elevenia. Halaman ini terdiri dari *form input* pencarian situs bukalapak, dari halaman 1 hingga ke *n*, namun yang disarankan adalah 10 halaman, hal ini bertujuan untuk meringankan kinerja *server*. Lalu *form input link URI* Elevenia dan pilihan kategori barang. Tampilan dari halaman *scraping* data barang dapat dilihat pada Gambar 5.7.

Scrape untuk ambil data

Kata Kunci: Mulai: Akhir:

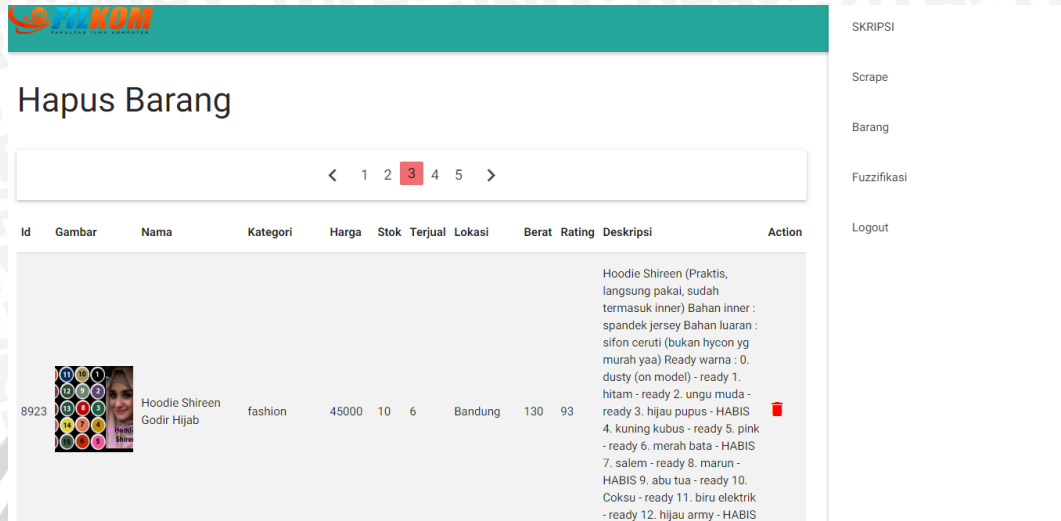
Kategori URI:

- SKRIPSI
- Scrape
- Barang
- Fuzzifikasi
- Logout

Gambar 5.7 Antarmuka Halaman Scraping Data Barang

5.4.8 Antarmuka Halaman Kelola Barang

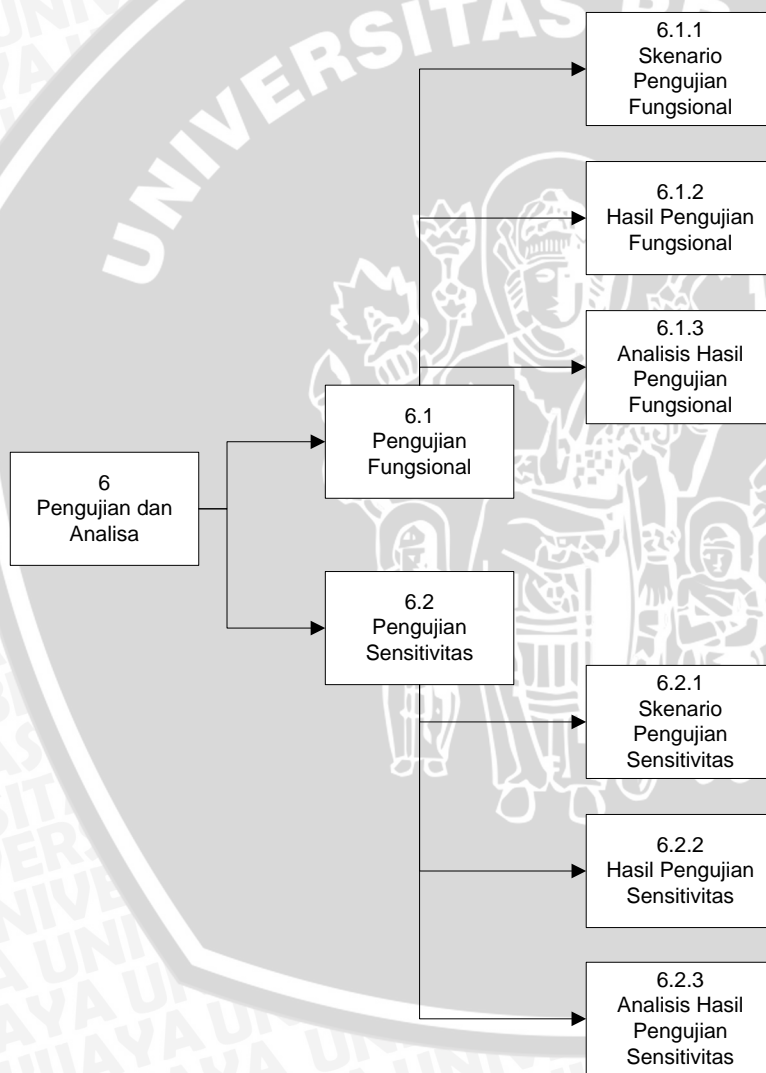
Halaman kelola barang adalah halaman yang digunakan *admin*, untuk menghapus barang yang tidak berhubungan lagi dengan sumber aslinya. Untuk melihat tampilan dari halaman kelola barang dapat dilihat pada Gambar 5.3.



Gambar 5.8 Antarmuka Halaman Kelola Barang

BAB 6 PENGUJIAN DAN ANALISA

Bab ini membahas proses pengujian dan analisa Sistem Pendukung Keputusan Rekomendasi Barang *Online* Menggunakan Metode *Web scraping dan Fuzzy Simple Additive Weighting*. Pengujian sistem dilakukan dengan cara melakukan pengujian fungsional menggunakan *blackbox testing*, pengujian tingkat akurasi dengan cara membandingkan data asli dan data hasil *scraping*, dan pengujian tingkat sensitivitas. Untuk lebih mudah memahami pengujian sistem, dapat dilihat pada diagram alir, pada Gambar 6.1.



Gambar 6.1 Diagram Alir Pengujian

Untuk melakukan pengujian sistem, dibutuhkan perancangan tabel uji. Adapun Tabel uji, sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scraping* dan *fuzzy-simple additive weighting*, adalah sebagai berikut:

1. Tabel pengujian fungsional

Tabel 6.1 Rancangan Tabel Pengujian Fungsional

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan

2. Tabel pengujian sensitivitas

Tabel 6.2 Rancangan Tabel Pengujian Sensitivitas

Total Nilai bobot	kriteria yang diuji	Jumlah Data	Data yang sesuai	Persentase kesesuaian

6.1 Pengujian Fungsional

Bagian ini berfungsi untuk menguji kebutuhan fungsional sistem, pengujian dilakukan dengan cara *blackbox testing* yaitu menguji masing-masing fungsi disetiap kelas uji atau setiap halaman. Data disajikan dalam bentuk tabel untuk lebih mudah dipahami.

6.1.1 Skenario Pengujian Fungsional

Skenario pengujian fungsional disesuaikan berdasarkan kelas uji, setiap skenario akan dilakukan pengujian, apakah sesuai dengan hasil yang diharapkan atau tidak sesuai.

6.1.2 Hasil Pengujian Fungsional

1. Pengujian halaman utama pencarian, disajikan dalam Tabel 6.3.

Tabel 6.3 Pengujian Halaman Utama Pencarian

Kelas Uji	Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
Halaman Utama Pencarian	Animasi <i>toggle</i> pada tombol rekomendasi	Sistem mampu menampilkan animasi <i>toggle</i> saat animasi diklik	Berhasil



	Saran Pencarian	Sistem dapat menampilkan saran data pencarian	Berhasil
--	-----------------	---	----------

2. Pengujian halaman rekomendasi, disajikan pada Tabel 6.4.

Tabel 6.4 Pengujian Halaman Rekomendasi

Kelas Uji	Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
Halaman Rekomendasi	Menampilkan 50 data teratas	Sistem menampilkan 50 data barang dengan nilai preferensi tertinggi.	Berhasil
	Mengganti angka urutan 1 hingga 3 dengan <i>image</i>	Sistem dapat mengganti angka pada urutan 1 hingga 3 dengan <i>image</i>	Berhasil
	Sorting rekomendasi barang	Sorting Rekomendasi barang berdasarkan nilai preferensi dari yang terbesar hingga terkecil	Berhasil

3. Pengujian halaman detail barang, disajikan pada Tabel 6.5.

Tabel 6.5 Pengujian Halaman Detail Barang

Kelas Uji	Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
Halaman Detail Barang	Detail barang	Sistem menampilkan detail informasi barang sesuai kriteria	Berhasil
	Logo sumber	Sistem menampilkan logo sumber berdasarkan <i>url</i> sumber	Berhasil

4. Pengujian halaman kategori barang, disajikan pada Tabel 6.6.

Tabel 6.6 Pengujian Halaman Kategori Barang

Kelas Uji	Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
Halaman Kategori Barang	Kategori barang	Sistem mampu menampilkan kategori dari hasil <i>scraping</i> barang	Berhasil
	Icon kategori	Menampilkan icon kategori setiap barang.	Berhasil

5. Pengujian halaman rekomendasi, disajikan pada Tabel 6.7.

Tabel 6.7 Pengujian Halaman Login

Kelas Uji	Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
Halaman Login	Pengisian <i>form</i> tanpa Username	Menampilkan pesan gagal saat tombol login ditekan	Berhasil
	Pengisian <i>form</i> tanpa Password	Menampilkan pesan gagal saat tombol login ditekan	Berhasil
	Form tidak diisi	Menampilkan pesan gagal saat tombol login ditekan	Berhasil
	Form diisi semua dan benar	Mengarahkan <i>admin</i> ke halaman dashboard	Berhasil
	Form diisi semua dan salah atau serangan SQL Injection	Menampilkan pesan gagal saat <i>form</i> diisi semua namun salah atau dengan <i>escape character</i>	Berhasil

6. Pengujian halaman kelola barang, disajikan pada tabel 6.6.

Tabel 6.8 Pengujian Halaman Kelola Barang

Kelas Uji	Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
Halaman Kelola Barang	Data barang dengan <i>pagination</i>	Berhasil mengarahkan <i>admin</i> kehalaman lain dengan data barang lain, sesuai batasan.	Berhasil
	Hapus barang	Menghapus satu data barang sesuai pilihan barang yang akan dihapus	Berhasil

7. Pengujian halaman *scraping* data, disajikan pada tabel 6.7.

Tabel 6.9 Pengujian Halaman *Scarping* Data

Kelas Uji	Skenario Uji	Hasil Yang Diharapkan	Kesimpulan
	Scraping Bukalapak	Sistem mampu mengambil data dari situs	Berhasil

Halaman Scraping Data		bukalapak, berdasarkan <i>keyword</i> pencarian	
	Scraping Elevenia	Sistem mampu mengambil data dari situs elevenia, berdasarkan <i>keyword</i> pencarian	Berhasil
	Pesan <i>update</i> atau <i>insert</i>	Menampilkan pesan berhasil di update atau di tambahkan	Berhasil

6.1.3 Analisa Hasil Pengujian Fungsional

Dari 19 skenario pengujian, 2 dinyatakan gagal karna belum dapat memenuhi harapan. Persentase tingkat keberhasilan dirumuskan pada persamaan 6-1.

$$Persentase = \frac{(semua\ skenario - skenario\ gagal)}{semua\ skenario} * 100\% \quad (6-1)$$

Dari persamaan 1 didapatkan nilai persentase sebesar $\frac{19}{19} * 100\% = 100\%$

6.2 Pengujian Sensitivitas

Pengujian sensitivitas adalah pengujian yang dilakukan terhadap respon sistem jika terjadi perubahan. Pengujian dilakukan dengan menguji nilai akurasi perangkingan data sebelum dan setelah terjadi perubahan.

6.2.1 Skenario Pengujian Sensitivitas

Pengujian sensitivitas dilakukan dengan cara menilai kinerja sistem jika terjadi perubahan. Data yang akan diubah pada pengujian sensitivitas ini adalah banyak data yang akan ditampilkan dan perubahan nilai bobot kriteria yang akan diuji. Perbandingan urutan adalah 100 data sesuai bobot kriteria normal dengan kata kunci yang berbeda.

6.2.2 Hasil Pengujian Sensivitas

Setelah dilakukan pengujian sensivitas maka didapatkan hasil pada tabel 6.10, 6.11 dan 6.12.

Tabel 6.10 Pengujian Sensivitas Kata Kunci Iphone 5

Total Nilai bobot	Total kriteria yang diuji	Jumlah Data (100)	Data yang sesuai	Persentase kesesuaian
1.0	7	10	1	10%
1.0	7	20	6	30%
1.0	7	30	11	36%
1.0	7	40	18	45%

1.0	7	50	25	50%
1.0	7	60	32	53%
1.0	7	70	42	56%
1.0	7	80	51	63%
1.0	7	90	74	82%
1.0	7	100	95	95%

Tabel 6.11 Sensivitas Kata Kunci Samsung

Total Nilai bobot	Total kriteria yang diuji	Jumlah Data (100)	Data yang sesuai	Persentase kesesuaian
1.0	7	10	2	10%
1.0	7	20	7	35%
1.0	7	30	13	43%
1.0	7	40	22	55%
1.0	7	50	31	62%
1.0	7	60	40	66%
1.0	7	70	49	70%
1.0	7	80	60	80%
1.0	7	90	76	84%
1.0	7	100	96	96%

Tabel 6.12 Pengujian Sensivitas Kata Kunci Sepatu

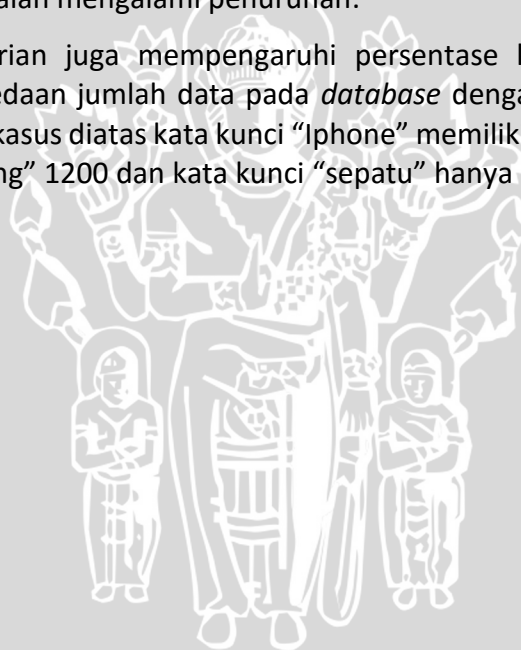
Total Nilai bobot	Total kriteria yang diuji	Jumlah Data (100)	Data yang sesuai	Persentase kesesuaian
1.0	7	10	5	50%
1.0	7	20	12	60%
1.0	7	30	19	63%
1.0	7	40	32	80%
1.0	7	50	41	82%
1.0	7	60	51	85%
1.0	7	70	62	88%

1.0	7	80	73	91%
1.0	7	90	84	93%
1.0	7	100	97	97%

6.2.3 Analisa Hasil Pengujian Sensitivitas

Kata kunci yang digunakan untuk pengujian sensitivitas diatas tidak sepenuhnya mewakili data, pengujian sensitivitas diatas bertujuan untuk melihat hubungan antara jumlah data dan nilai presentase kesesuaian. Dari hasil pengujian maka didapatkan analisa sebagai berikut:

- Jika jumlah barang yang akan diuji mendekati nilai target uji (dalam kasus diatas adalah 100 data yang dijadikan *sample* uji), maka persentase kesesuaian mengalami peningkatan.
- Semakin sedikit data barang yang akan diuji dan menjauhi target uji, maka persentase kesesuaian mengalami penurunan.
- Kata kunci pencarian juga mempengaruhi persentase kesesuaian hal ini dikarenakan perbedaan jumlah data pada *database* dengan kata kunci yang ditargetkan, pada kasus diatas kata kunci "Iphone" memiliki jumlah data 2140, kata kunci "samsung" 1200 dan kata kunci "sepatu" hanya 210 data.



BAB 7 PENUTUP

Bab ini membahas mengenai kesimpulan dan saran yang didapatkan dari hasil penelitian yang telah dilaksanakan.

7.1 Kesimpulan

E-commerce dapat menjadi penghubung tanpa mengenal jarak, semakin banyak kecendrungan masyarakat untuk berbisnis melalui *E-commerce* maka semakin besar peluang terjadinya tindak kejahatan, penipuan dan sebagainya. Dengan cara merekomendasikan barang terbaik dari seluruh situs *E-commerce* diharapkan mampu meminimalisir tindak kejahatan *E-commerce*.

Berdasarkan hasil perancangan, implementasi, pengujian dan analisis didapatkan kesimpulan sebagai berikut:

1. Rekomendasi barang menggunakan metode *web scraping* dan *fuzzy simple additive weighting* berhasil diterapkan untuk merekomendasikan barang terbaik berdasarkan kriteria yang menjadi pertimbangan.
2. Akurasi fungsional sistem mencapai 100% artinya bahwa sistem berjalan baik dan sesuai dengan perancangan.
3. Dari hasil uji sensitivitas didapatkan kesimpulan bahwa, semakin sedikit jumlah barang yang akan direkomendasikan maka akurasi kesesuaian ranking semakin kecil. Sebaliknya, semakin banyak jumlah barang, maka kesesuaian ranking semakin akurat (berdasarkan 100 data uji).
4. Kata kunci mempengaruhi nilai presentase kesesuaian pada uji sensitivitas, dikarenakan jumlah data uji keseluruhan mempengaruhi nilai preferensi pada perankingan.

7.2 Saran

Saran yang didapatkan berdasarkan pengembangan aplikasi sistem pendukung keputusan rekomendasi barang *online* menggunakan metode *web scraping* dan *fuzzy-simple additive weighting* adalah sebagai berikut:

1. Data yang ditampilkan sesuai dengan data aslinya, namun hanya pada saat dilakukan proses *scraping*, perubahan data dapat terjadi setiap saat, terutama pada kriteria stok barang. Disarankan jika memiliki akses data API lebih baik menggunakan data API untuk mendapatkan data secara *real time*.
2. Data dalam format JSON dan untuk mengaksesnya membutuhkan *access-token* sulit untuk diambil. Sehingga dibutuhkan algoritma/program khusus yang mampu mendapatkan data JSON dan membutuhkan *access-token*.

DAFTAR PUSTAKA

- Alexa, (2016). *Bukalapak.com traffic status and site overview*. Tersedia di:<<http://www.alexacom/siteinfo/bukalapak.com>> [Diakses 18 February 2016]
- Alexa, (2016). *Elevenia.co.id traffic status and site overview*. Tersedia di:<<http://www.alexacom/siteinfo/elevenia.co.id>> [Diakses 18 February 2016]
- Boettcher I., 2015. *Automatic data collection on the internet (web scraping)*. [online] Tersedia di:< <https://ec.europe.eu> > [Diakses 22 February 2016]
- Fernandez J.I, Blasco J., Iglesias C.A., Garijo M., 2009. *A semantic scraping for web resources Applying Linked Data to Web Page Screen Scraping*. [online] Tersedia di:< <https://www.researchgate.net> > [Diakses 23 February 2016]
- Kaho, M.L.R, Tanaamah,A.R, dan Wowor A.D., 2013. *simple additive weighting pada penentuan tingkat kesejahteraan penduduk propinsi Nusa Tenggara Timur*. [online] Tersedia di:< <http://repository.uksw.edu> > [Diakses 20 February 2016]
- Kusumadewi, Sri., Hartati, S., Harjoko, A., dan Wardoyo, R. (2006). *Fuzzy Multi-Attribute Decision Making (FUZZY MADM)*. Yogyakarta: Penerbit Graha Ilmu.
- Rinaldhi, G.E., 2013. Penerapan Metode Simple additive weighting (SAW) Untuk Sistem Pendukung Keputusan Penentuan Penerimaan Beasiswa Bantuan Siswa Miskin (BSM) Pada SMA Negeri 1 Subah Kab.Batang. [online] Tersedia di:< <http://eprints.dinus.ac.id/5406/> > [Diakses 20 February 2016]
- Subakti, I, 2002. *Sistem Pendukung Keputusan (Decision Support System)*. Jurusan Teknik Infomatika Fakultas Teknologi Informasi. Intitut Teknologi Sepuluh November. Surabaya
- Tim Pengampu SPK Ganjil 2013., 2013. Pengantar DSS & Management Support System. [ppt] Universitas Brawijaya.
- Usto, N.J., 2013. *Sistem Pendukung Keputusan Penilaian Proses Belajar Mengajar Menggunakan Metode Simple Additive Weighting (SAW)*. S2. Universitas Diponegoro.
- Yohana D.L.W., Maya S.R, Rachmawati H., 2013. *Sistem Pendukung Keputusan Penentuan Karyawan Terbaik Menggunakan Metode Saw (Simple additive weighting) Studi Kasus Pt. Pertamina Ru li Dumai*. [online] Tersedia di:< <http://yohana.komputer.pcr.ac.id> > [Diakses 20 February 2016]

LAMPIRAN – LAMPIRAN

Lampiran 1

Data Barang Bukalapak Diakses 25 Maret 2016

Nomor	Alternatif	Kriteria						
		C1	C2	C3	C4	C5	C6	C7
1	Iphone 5 32gb black lte	0,1567	0,2778	0,0400	0,2778	0,2708	0,4567	0,0400
2	iPhone 5s (16GB) - Space Grey - Garansi Resmi TAM	0,1567	0,2778	0,0400	0,1111	0,1458	0,4567	0,0400
3	iphone 5S SILVER 64GB GARANSI RESMI 1TAHUN NEW	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,0400
4	iphone 5 GOLD 16GB (NEW GARANSI 1TAHUN) PROMO	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,0400
5	Iphone 5S/ Iphone 5G Transparan Case Flipcase Flipcover	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,0900
6	iPhone 5S 64GB white / black Garansi Distributor 1thn	0,2567	0,2778	0,0900	0,1111	0,2708	0,4567	0,0400
7	IPHONE 5 WHITE (FREE ONGKIR Se Jabodetabek)	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,0400
8	iphone 5 32GB (NEW GARANSI 1 TAHUN) PROMO	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,0400
9	iphone 5S 32 GOLD (NEW GARANSI 1 TAHUN)	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,0900
10	IPHONE 5 64GB SILVER - GARANSI 1 TAHUN DISTRIBUTOR	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,2567
11	IPHONE 5 32GB WHITE GOLD GARANSI PLATINUM 1 Tahun (Putih Gold))	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,1567
12	Iphone 5 32GB HITAM Garansi Distributor Platinum 1 Tahun	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,0900
13	iphone 5 32gb Black n White	0,1567	0,2778	0,1567	0,1111	0,2708	0,4567	0,1567
14	Iphone 5 16gb Black n White	0,1567	0,2778	0,1567	0,1111	0,2708	0,4567	0,1567

15	IPHONE 5 16GB Putih ORIGINAL Garansi Platinum 1 Tahun	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,0900
16	Iphone 5 16GB Black Garansi Platinum 1 Tahun	0,1567	0,2778	0,1567	0,1111	0,2708	0,4567	0,1567
17	Iphone 5s 16gb Space Gray Mulus Like New Second Original Fullset Non Jepang	0,1567	0,1111	0,0400	0,2778	0,2708	0,4567	0,0400
18	Iphone 5S 16GB Hitam Garansi Distributor 1 Tahun Platinum	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,1567
19	IPHONE 5S 16GB Grey GARANSI PLATINUM 1TH	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,1567
20	IPHONE 5 64GB BLACK - GARANSI 1 TAHUN DISTRIBUTOR	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,1567

Lampiran 2

Data Barang Elevenia Diakses 25 Maret 2016

Nomor	Alternatif	Kriteria						
		C1	C2	C3	C4	C5	C6	C7
1	iPhone 5S 32GB	0,1567	0,1111	0,0900	0,6111	0,2708	0,4567	0,1567
2	Iphone 5s 16gb Grey / space gray	0,1567	0,2778	0,1567	0,1111	0,2708	0,4567	0,1567
3	iphone 5s 64gb space gray / grey	0,1567	0,2778	0,1567	0,1111	0,2708	0,4567	0,1567
4	iphone 5c original 16gb garansi 12 bln	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,0400
5	Iphone 5 32GB White Garansi Platinum 1 tahun	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,1567
6	IPHONE 5 16GB GOLD	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,0900
7	IPHONE 5 32GB BLACK GOLD GARANSI DISTRIBUTOR PLATINUM (HITAM GOLD)	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,1567
8	iphone 5s 64gb silver	0,1567	0,1111	0,0400	0,1111	0,2708	0,4567	0,0900
9	IPHONE 5 32GB SILVER - GARANSI 1 TAHUN DISTRIBUTOR	0,1567	0,2778	0,1567	0,1111	0,2708	0,4567	0,2567
10	iphone 5 white 16gb	0,1567	0,1111	0,0400	0,1111	0,2708	0,4567	0,0900
11	iPhone 5S 16GB GOLD	0,1567	0,1111	0,0900	0,6111	0,2708	0,4567	0,1567
12	iPhone 5 16GB Hitam/Putih	0,1567	0,1111	0,0900	0,6111	0,2708	0,4567	0,1567
13	Iphone 5s 32GB Gold Garansi Platinum 1 thn	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,0400
14	iphone 5s original 64gb gold garansi 12 bln	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,0400
15	Iphone 5s 16gb Gold Distributor 1 tahun	0,1567	0,2778	0,1567	0,1111	0,2708	0,4567	0,1567
16	Iphone 5S 32GB GSM Hitam Chrome Garansi Distributor 1 Tahun	0,1567	0,2778	0,0900	0,1111	0,2708	0,4567	0,0900
17	Iphone 5 32GB Black Gold Garansi Platinum 1 Tahun	0,1567	0,2778	0,0400	0,1111	0,2708	0,4567	0,1567

18	IPHONE 5s 16GB SPACE GRAY - GARANSI 1 TAHUN DISTRIBUTOR	0,1567	0,2778	0,1567	0,1111	0,2708	0,4567	0,1567
19	IPHONE 5s 32GB GOLD - GARANSI 1 TAHUN DISTRIBUTOR	0,1567	0,2778	0,1567	0,1111	0,2708	0,4567	0,1567
20	iphone 5s gold 16gb	0,1567	0,1111	0,0400	0,1111	0,2708	0,4567	0,0900

