

**MESIN PENCARI BERBASIS *SEMANTIC WEB* MENGGUNAKAN
ALGORITMA BOYER-MOORE PADA ENSIKLOPEDIA TANAMAN
OBAT**

SKRIPSI

Oleh:

**FAIQUL IHSAN
NIM. 09650076**



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2013**

**MESIN PENCARI BERBASIS *SEMANTIC WEB* MENGGUNAKAN
ALGORITMA BOYER-MOORE PADA ENSIKLOPEDIA TANAMAN
OBAT**

SKRIPSI

Diajukan Kepada:

**Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

Oleh:

**FAIQUL IHSAN
NIM. 09650076**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2013**

**MESIN PENCARI BERBASIS *SEMANTIC WEB* MENGGUNAKAN
ALGORITMA BOYER-MOORE PADA ENSIKLOPEDIA TANAMAN
OBAT**

SKRIPSI

Oleh:

**FAIQUL IHSAN
NIM. 09650076**

Telah disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

**A'la Syauqi, M.Kom
NIP. 19771201 200801 1 007**

**Fatchurrochman, M.Kom
NIP. 19700731 200501 1 002**

Malang, 12 September 2013

**Mengetahui
Ketua Jurusan Teknik Informatika**

**Dr. Cahyo Crysdian
NIP. 19720309 200501 2 002**

**MESIN PENCARI BERBASIS *SEMANTIC WEB* MENGGUNAKAN
ALGORITMA BOYER-MOORE PADA ENSIKLOPEDIA TANAMAN
OBAT**

SKRIPSI

Oleh:
FAIQUL IHSAN
NIM. 09650076

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 19 September 2013

Susunan Dewan Penguji:	Tanda Tangan
1. Penguji Utama : Irwan Budi Santoso, M.Kom NIP. 19770103 201101 1 004	()
2. Ketua : Linda Salma Angreani, MT NIP. 19770803 200912 2 005	()
3. Sekretaris : A'la Syauqi, M.Kom NIP. 19771201 200801 1 007	()
4. Anggota : Fatchurrochman, M.Kom NIP. 19700731 200501 1 002	()

Mengetahui dan Mengesahkan,
Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

SURAT PERNYATAAN ORISINALITAS PENELITIAN

Saya yang bertanda tangan di bawah ini :

Nama : FAIQUL IHSAN

NIM : 09650076

Fakultas / Jurusan : SAINS DAN TEKNOLOGI / TEKNIK INFORMATIKA

Judul Penelitian : MESIN Pencari Berbasis *SEMANTIC WEB*
MENGUNAKAN ALGORITMA BOYER-MOORE
PADA ENSIKLOPEDIA TANAMAN OBAT

Menyatakan dengan sebenar-benarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur jiplakan, maka saya bersedia untuk mempertanggung jawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 12 September 2013

Yang Membuat Pernyataan,

FAIQUL IHSAN
NIM. 09650076

MOTTO

قُلْ سِيرُوا فِي الْأَرْضِ فَانظُرُوا كَيْفَ بَدَأَ الْخَلْقَ ثُمَّ اللَّهُ يُنشِئُ
النَّشْأَةَ الْآخِرَةَ إِنَّ اللَّهَ عَلَىٰ كُلِّ شَيْءٍ قَدِيرٌ ﴿٢٠﴾

"Katakanlah : Berjalanlah di muka bumi, maka perhatikanlah bagaimana (Allah) memulai penciptaan (makhluk), kemudian Allah menjadikan kejadian yang akhir, sesungguhnya Allah Maha Kuasa atas segala sesuatu"

"Lihat, Pelajari, Amalkan"

Dengan mengucapkan syukur Alhamdulillah,
karya sederhana ini ku persembahkan untuk orang-orang yang telah memberi
warna dalam hidupku.

Ibu dan Bapakku. Terima kasih telah membersarkan, mendidik, memberikan
kasih sayang serta do'a dan dukungannya, Semangad kaliand selalu menyertai
jalanku.

Adik-adikku : Hilya, U'ul, Bagus, Atsmar dan didil. Terima kasih atas
dukungan dan doa kaliand

Teman-teman Tim Semantik : Ayi, Andy, Dian & Ludvi. Terima kasih atas
kerjasamanya hingga penelitian ini bisa mencapai hasil seperti hari ini.

Teman-teman Tim Kasembon Beruntung : Dian Ayu, Firda, Fatiyah Dkk.
Terima kasih atas semua bantuan, motivasi serta semangad yang telah kaliand
berikan selama ini.

Teman-teman kost-an : Ilmi, jack, choesnan, mas muslim dan ipul . Terima
kasih atas supportnya, hingga Q tetap melangkah dan dapat menyelesaikan
penelitian ini.

KATA PENGANTAR

Segala puji bagi Allah SWT karena atas rahmat, taufiq dan hidayah-Nya, penulis dapat menyelesaikan penulisan tugas akhir sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer (S.Kom). Penulis menyadari bahwa banyak pihak yang telah membantu dalam menyelesaikan penulisan tugas akhir ini.

Untuk itu, iringan doa dan ucapan terima kasih yang sebesar-besarnya penulis sampaikan kepada:

1. Prof. Dr. H. Mudjia Rahardjo, M.Si, Rektor Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
2. Dr.drh. Bayyinatul Muchtaromah, M.Si, Dekan Fakultas Sains dan Teknologi Maulana Malik Ibrahim Malang.
3. Dr. Cahyo Crysdian, Ketua Jurusan Teknik Informatika Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang.
4. A'la Syauqi, M.Kom, selaku pembimbing satu karena atas bimbingan, pengarahan, dan kesabarannya penulisan tugas akhir ini dapat terselesaikan.
5. Fatchurrochman, M.Kom, selaku pembimbing dua karena atas bimbingan dan nasehatnya dalam penulisan tugas akhir ini.
6. M. Ainul Yaqin, S.Si, M.Kom, selaku dosen wali yang selalu membimbing dan mengarahkan dalam merencanakan studi.
7. Seluruh Dosen Teknik Informatika UIN Maulana Malik Ibrahim Malang, yang telah memberikan ilmu dan pengetahuan sebagai bekal dalam menyelesaikan tugas akhir.
8. Bapak, Ibu, dan adik-adikku tercinta yang dengan sepenuh hati memberikan dukungan moral maupun spiritual sehingga penulisan tugas akhir dapat terselesaikan.
9. Teman-teman dari Jurusan Teknik Informatika Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang yang telah memberikan dukungan, motivasi, serta informasi dalam menyelesaikan tugas akhir.

Atas segala kekurangan dalam laporan ini, penulis memohon maaf dan mengharap kritik serta saran dari pembaca. Semoga tugas akhir ini dapat bermanfaat dan menambah khasanah ilmu pengetahuan.

Malang, 12 September 2013

Penulis



DAFTAR ISI

Halaman Judul.....	i
Lembar Pengajuan.....	ii
Lembar Persetujuan.....	iii
Lembar Pengesahan	iv
Surat Pernyataan.....	v
Motto.....	vi
Lembar Persembahan	vii
Kata Pengantar	viii
Daftar Isi.....	x
Daftar Tabel	xiii
Daftar Gambar.....	xiv
ABSTRAK.....	xvi
ABSTRACT.....	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian.....	4
1.4 Manfaat Penelitian.....	4
1.5 Batasan Masalah.....	5
1.6 Metodologi Penelitian	5
1.7 Sistematika Penulisan.....	6
BAB II TINJAUAN PUSTAKA	8
2.1 Ensiklopedia	8
2.2 Tanaman Obat.....	9
2.2.1 Bagian-Bagian Tumbuhan	10
2.2.1.1 Alat Hara.....	11
2.2.1.2 Daun.....	11
2.2.1.3 Batang	11
2.2.1.4 Akar	11
2.2.1.5 Bagian-Bagian Lain	11
2.2.1.6 Alat Perkembangbiakan.....	12
2.2.1.7 Bunga.....	12
2.2.1.8 Buah.....	13
2.2.1.9 Biji	13
2.2.2 Tumbuhan yang berkhasiat.....	13
2.2.2.1 Semangka.....	13
2.2.2.2 Bawang Merah.....	13
2.2.2.3 Terong.....	13
2.2.2.4 Jintan Hitam.....	14
2.2.2.5 Anggur	14

2.3	<i>Semantic</i>	14
2.3.1	<i>Semantic web</i>	14
2.3.2	<i>Semantic search</i>	19
2.4	Ontologi	20
2.5	<i>Platform Android</i>	20
2.6	Algoritma Boyer-Moore	22
2.7	<i>Jena Framework</i>	26
2.8	<i>Protégé</i>	28
2.9	<i>Smore</i>	29
BAB III	ANALISIS DAN PERANCANGAN	30
3.1	Sumber Data Tanaman	31
3.2	Tahapan <i>Semantic Web</i>	34
3.3	Perancangan Aplikasi	36
3.3.1	Desain Aplikasi	36
3.3.2	Desain Interface	39
3.3.2.1	Halaman Awal Administrator	40
3.3.2.2	Halaman Data Tanaman	41
3.3.2.3	Halaman Data Khasiat	42
3.3.2.4	Halaman Data Bagian Tanaman	43
3.3.2.5	Halaman Data Bagian Digunakan	44
3.3.2.6	Halaman Data Cara Pemakaian	45
3.3.2.7	Halaman Data Penelitian	46
3.3.2.8	<i>Interface</i> Awal Aplikasi <i>Mobile</i>	47
3.3.2.9	<i>Interface</i> Pencarian	48
3.3.2.10	<i>Interface</i> Data A-Z	49
3.4	Perancangan Ontologi	50
3.4.1	Rancangan Domain Dan Ruang Lingkup Ontologi	51
3.4.2	Mendefinisikan Kelas Dan Tingkat Kelas	51
3.4.3	Mendefinisikan Properti Kelas Dan Faset Dari S _{Lot}	52
3.4.4	Membuat <i>Instance</i>	56
3.4.5	Skema <i>Graph Ontology</i>	57
3.5	<i>Crawling</i> Menggunakan Sphider	58
3.6	<i>Indexing</i> Kata Dari Halaman Website	58
3.7	<i>Markup</i> Halaman Website Menggunakan Smore	59
3.8	Menabelkan N-Triple Menggunakan API JENA	61
3.9	Proses Pencarian	61
3.10	Proses Pencocokan Dengan Metode Boyer-Moore	63
3.11	Rancang Uji Coba Aplikasi	70
3.12	Kebutuhan Sistem	70
3.12.1	Perangkat Untuk Pembuatan Aplikasi	70
3.12.2	Perangkat Untuk Mejalankan Aplikasi	71
BAB IV	HASIL DAN PEMBAHASAN	72
4.1	Aplikasi Halaman Admin Menggunakan PHP	72
4.1.1	Halaman Awal Administrator	72

4.1.2	Halaman Data Tanaman Obat.....	73
4.1.3	Halaman <i>Input</i> Tanaman.....	74
4.1.4	Halaman Detail Tanaman Obat.....	75
4.1.5	Halaman Data Penelitian	75
4.1.6	Halaman <i>Input</i> Penelitian	76
4.1.7	Halaman Detail Penelitian	77
4.1.8	Halaman Data Khasiat	77
4.1.9	Halaman <i>Input</i> Khasiat	78
4.1.10	Halaman Data Bagian	78
4.1.11	Halaman <i>Input</i> Bagian	79
4.1.12	Halaman Kegunaan Bagian Tanaman.....	79
4.1.13	Halaman <i>Input</i> Kegunaan Bagian Tanaman	80
4.1.14	Halaman Data Pemakaian.....	81
4.1.15	Halaman <i>Input</i> Pemakaian	82
4.2	<i>Preprocessing</i> Pencarian	82
4.2.1	<i>Crawling</i>	82
4.2.2	<i>Indexing</i>	84
4.2.3	Pembuatan Metadata.....	85
4.3	Proses Pencarian.....	88
4.3.1	Menyaring Kata Penting Dari Kalimat Pencarian	88
4.3.2	Pencarian Kata Pada Tabel <i>Indexing</i>	89
4.3.3	Pencarian Kata Pada Tabel <i>Ntriple</i>	89
4.3.4	Pencocokan Kata Hasil Indexing dan Kata Hasil <i>Ntriple</i> Dengan Menggunakan Metode Boyer-Moore.....	90
4.3.5	Merubah Hasil Pencarian Ke Dalam Bentuk J-SON.....	91
4.4	Proses Pencarian Menggunakan Metode Boyer-Moore	91
4.4.1	Perhitungan <i>Bas Character</i>	91
4.4.2	Perhitungan <i>Good-Suffikx</i>	92
4.5	Implementasi <i>Interface</i> Pada <i>Mobile</i>	93
4.5.1	<i>Interface</i> Awal Pada Aplikasi.....	93
4.5.2	<i>Interface List</i> Pencarian	93
4.5.3	<i>Interface</i> Pencarian	94
4.5.4	<i>Interface</i> Hasil Pencarian.....	95
4.5.5	Detail Hasil Pencarian	95
4.6	Uji Coba Aplikasi	96
4.7	Integrasi Keilmuan	99
BAB V	KESIMPULAN DAN SARAN.....	103
5.1	Kesimpulan.....	103
5.2	Saran	103
	DAFTAR PUSTAKA	104

DAFTAR TABEL

Tabel 2.1	<i>Occurence Heuristic</i>	24
Tabel 2.2	<i>Match Heuristic</i>	25
Tabel 2.3	Pergeseran pertama	25
Tabel 2.4	Pergeseran kedua.....	25
Tabel 2.5	Pergeseran ketiga.....	25
Tabel 2.6	Pergeseran keempat.....	26
Tabel 2.7	Pergeseran kelima	26
Tabel 3.1	Atlas Tumbuhan Obat Jilid 3	32
Tabel 3.2	Tabel tanaman	37
Tabel 3.3	Tabel bagian_tanaman.....	38
Tabel 3.4	Tabel bagian_digunakan	38
Tabel 3.5	Tabel khasiat	38
Tabel 3.6	Tabel cara_pemakaian.....	38
Tabel 3.7	Tabel penelitian	39
Tabel 3.8	Tabel user	39
Tabel 3.9	Rancangan properti pada <i>class</i> tanaman	53
Tabel 3.10	Rancangan properti pada <i>class</i> penelitian	53
Tabel 3.11	Rancangan properti pada <i>class</i> pemakaian.....	53
Tabel 3.12	Rancangan properti pada <i>subclass</i> asing <i>class</i> nama	53
Tabel 3.13	Rancangan properti pada <i>subclass</i> daerah <i>class</i> nama	53
Tabel 3.14	Rancangan properti pada <i>subclass</i> latin <i>class</i> nama.....	54
Tabel 3.15	Rancangan properti pada <i>subclass</i> simplisia <i>class</i> nama	54
Tabel 3.16	Rancangan properti pada <i>subclass</i> sinonim <i>class</i> nama.....	54
Tabel 3.17	Rancangan properti pada <i>subclass</i> suku <i>class</i> nama	54
Tabel 3.18	Rancangan properti <i>class</i> khasiat	54
Tabel 3.19	Rancangan properti <i>class</i> kandungan_kimia.....	54
Tabel 3.20	Rancangan properti pada <i>class</i> bagian_tanaman.....	54
Tabel 3.21	Pembentukan <i>suffix</i>	66
Tabel 3.22	Tabel pergeseran OH dan MH.....	66
Tabel 3.23	Tabel <i>iterasi</i> pertama.....	67
Tabel 3.24	Tabel <i>iterasi</i> kedua.....	67
Tabel 3.25	Tabel <i>iterasi</i> ketiga.....	68
Tabel 3.26	Tabel <i>iterasi</i> keempat.....	68
Tabel 3.27	Tabel <i>iterasi</i> kelima.....	69
Tabel 4.1	Uji coba perbandingan kecepatan	96
Tabel 4.2	Uji coba perbandingan keakuratan.....	97
Tabel 4.3	Perbandingan keakuratan dan kecepatan rata-rata	98

DAFTAR GAMBAR

Gambar 2.1	Arsitektur <i>Semantic Web</i>	16
Gambar 2.2	<i>Patern Boyer-Moore</i>	22
Gambar 2.3	<i>Pergeseran Patern Boyer-Moore</i>	22
Gambar 2.4	Arsitektur Jena.....	28
Gambar 2.5	Smore Web Page	29
Gambar 3.1	Rancangan aplikasi tahap <i>preprocessing</i>	30
Gambar 3.2	Rancangan aplikasi tahap <i>processing</i>	31
Gambar 3.3	Tahapan proses dalam membangun <i>semantic web</i>	35
Gambar 3.4	Rancangan tampilan awal.....	40
Gambar 3.5	Rancangan <i>input</i> data tanaman.....	41
Gambar 3.6	Rancangan <i>view</i> data tanaman.....	42
Gambar 3.7	Rancangan <i>input</i> data khasiat	42
Gambar 3.8	Rancangan <i>view</i> data khasiat.....	43
Gambar 3.9	Rancangan <i>input</i> bagian tanaman.....	43
Gambar 3.10	Rancangan <i>view</i> data bagian tanaman	44
Gambar 3.11	Rancangan <i>input</i> data bagian digunakan	44
Gambar 3.12	Rancangan <i>view</i> data bagian digunakan.....	45
Gambar 3.13	Rancangan <i>input</i> data pemakaian	45
Gambar 3.14	Rancangan <i>view</i> data pemakaian	46
Gambar 3.15	Rancangan input data penelitian.....	46
Gambar 3.16	Rancangan <i>view</i> data penelitian.....	47
Gambar 3.17	Rancangan tampilan awal	47
Gambar 3.18	Rancangan tampilan cari tanaman.....	48
Gambar 3.19	Rancangan tampilan hasil cari.....	48
Gambar 3.20	Rancangan tampilan menu data tanaman A-Z.....	49
Gambar 3.21	Rancangan tampilan data tanaman A-Z	50
Gambar 3.22	Rancangan tampilan informasi tanaman	50
Gambar 3.23	Rancangan Pendefinisian <i>Class</i> dan Tingkatan <i>Class</i>	52
Gambar 3.24	Properti-Properti yang terbentuk pada <i>Subclass</i> penelitian.....	55
Gambar 3.25	Pendefinisian Fungsi <i>Inverse</i> pada Tiap <i>Object Property</i>	56
Gambar 3.26	Individual-Individual yang Terdapat pada tanaman.....	57
Gambar 3.27	Skema <i>Ontology</i> Tanaman Obat.....	57
Gambar 3.28	Proses Alur Crawling pada spider	58
Gambar 3.29	Proses Alur Indexing Halaman web	59
Gambar 3.30	Alur <i>Proses Smore</i>	60
Gambar 3.31	Alur Proses Penabelan N-triple	61
Gambar 3.32	Alur <i>Proses</i> pencarian	62
Gambar 3.33	Alur Pembentukan Tabel <i>Occurence Heuristic</i>	63
Gambar 3.34	Alur Pembentukan Tabel <i>Match Heuristic</i>	64
Gambar 3.35	Alur Pencocokan dengan Algoritma Boyer Moore	65
Gambar 3.36	Rancangan uji coba aplikasi	70
Gambar 4.1	Tampilan depan halaman administrator	73
Gambar 4.2	Tampilan data tanaman obat.....	73

Gambar 4.3	Tampilan <i>input</i> tanaman	74
Gambar 4.4	Tampilan detail tanaman obat	75
Gambar 4.5	Tampilan data penelitian	76
Gambar 4.6	Tampilan <i>input</i> penelitian.....	76
Gambar 4.7	Tampilan detail penelitian	77
Gambar 4.8	Tampilan data khasiat	77
Gambar 4.9	Tampilan <i>input</i> khasiat	78
Gambar 4.10	Tampilan data bagian tanaman	79
Gambar 4.11	Tampilan <i>input</i> bagian	79
Gambar 4.12	Tampilan data kegunaan bagian tanaman.....	80
Gambar 4.13	Tampilan <i>input</i> kegunaan bagian tanaman	81
Gambar 4.14	Tampilan data cara pemakaian tanaman.....	81
Gambar 4.15	Tampilan input cara pemakaian.....	82
Gambar 4.16	Tampilan http://tobat.web-semantik.com	83
Gambar 4.17	Tampilan aplikasi Sphider untuk untuk proses <i>Crawling</i>	83
Gambar 4.18	Tampilan hasil <i>Crawling</i>	84
Gambar 4.19	<i>Source Code Parsing title</i> dan content pada <i>tag body</i>	84
Gambar 4.20	<i>Source</i> perhitungan kata dan <i>insert</i> kedalam tabel indexing	85
Gambar 4.21	<i>Interface Proses Indexing</i>	85
Gambar 4.22	<i>Triple Table</i> hasil pemberian metadata	86
Gambar 4.23	Hasil metadata dengan menggunakan Smore.....	86
Gambar 4.24	<i>Source code</i> untuk membaca <i>N-Triple</i>	87
Gambar 4.25	<i>Source code parsing</i> SPO	87
Gambar 4.26	Tampilan aplikasi untuk membaca dan memasukkan SPO kedalam database	87
Gambar 4.27	<i>Source code</i> penyaringan kata	88
Gambar 4.28	<i>Source code</i> pencocokan kata pada tabel indexing	89
Gambar 4.29	<i>Source code</i> pencocokan kata pada tabel ntriple	90
Gambar 4.30	<i>Source code</i> pencocokan hasil indexing dan ntriple dengan metode Boyer-Moore	90
Gambar 4.31	<i>Source code</i> merubah hasil menjadi bentuk J-SON	91
Gambar 4.32	<i>Source code</i> perhitungan <i>bad character</i>	92
Gambar 4.33	<i>Source code</i> perhitungan <i>good suffix</i>	92
Gambar 4.34	<i>interface</i> awal user.....	93
Gambar 4.35	<i>interface</i> list semua data tanaman obat.....	94
Gambar 4.36	<i>Interface</i> pencarian tanaman obat.....	94
Gambar 4.37	<i>Interface</i> hasil pencarian.....	95
Gambar 4.38	<i>Interface</i> detail pencarian tanaman obat	95

ABSTRAK

Ihsan, Faiqul. 2013. **Mesin Pencari Berbasis *Semantic Web* Menggunakan Algoritma Boyer-Moore pada Ensiklopedia Tanaman Obat.** Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) A'la Syauqi, M.Kom, (II) Fatchurrochman, M.Kom.

Kata Kunci : Boyer-Moore, *Crawling*, *Indexing*, Ontologi, SPO, *Semantic Search*

Indonesia memiliki keanekaragaman jenis tanaman yang memiliki jumlah spesies 30.000 dan yang berpotensi sebagai obat sekitar 9.000 spesies tanaman. Namun manfaat dari tanaman yang berpotensi sebagai obat kurang dikenal oleh masyarakat dikarenakan kurangnya dokumentasi, sehingga banyak penelitian di bidang botani dan obat tradisional berlomba-lomba membuat sebuah dokumentasi mengenai tanaman obat. Karena banyaknya dokumentasi yang dibuat oleh para peneliti, maka dibutuhkan sebuah mesin pencarian yang akurat. Penelitian ini menerapkan konsep *semantic search*. Pada mesin pencariannya memiliki beberapa proses, yaitu *crawling*, *indexing*, membangun ontologi, pemetadataan ontologi menggunakan *Smore*. Metadata dalam bentuk *triple table* akan dilabelkan dalam bentuk SPO (Subjek, Predikat, Objek) dengan menggunakan *library* API Jena. Algoritma Boyer-Moore digunakan sebagai *string matching* pada proses pencarian aplikasi ini. Aplikasi ensiklopedia tanaman obat di Indonesia dibangun untuk *smartphone* bersistem operasi Android. Berdasarkan hasil pengujian aplikasi memiliki keakuratan mencapai 95%, dalam melakukan pencarian dari kalimat pencarian yang dimasukkan oleh *user*.

ABSTRACT

Ihsan, Faiqul. 2013. **Semantic Web-Based Search Engines Using Boyer-Moore algorithm in Encyclopedia of Medicinal Plants**. Infomatika Engineering Department Faculty of Science and Technology of the State Islamic University Maulana Malik Ibrahim Malang. Supervisor : (I) A'la Syauqi, M.Kom, (II) *Fatchurrochman, M.Kom*.

Keyword : Boyer-Moore, Crawling, Indexing, Ontology, SOP, Semantic Search

Indonesia has there are many plants that have 30.000 specieses and for about 9000 specieses are medicine,however the function of plants that have medicine potencial is not recognized enough by public it caused by less of documentation, until there are many research area in botani and traditional medicine compete to make documentation about medicine plants,because there are many documentation that is made by scientist ,so it's needed a accurate mechine of searching.this experiment apply semantic, search draft.in the search mechine has many process,namely crawling, indexing, ontology making mapdata ontology by use softwere smore. Then from yield metadata that shaped triple table will be done table SOP (subject,predicat,object) with library API jena helping. Algorithm is used to verification string in this application is Boyer Moore algorithm.this application is applied of ensiklopedia medicine plant in Indonesia that specially to give information about Indonesia medicine plant make use of smartphone technology by Android operation system. According the yield testing can has accurate reach 95% of searching from sentence search thas is entered by user.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada masa ini obat-obatan yang beredar kebanyakan mengandung bahan kimia yang bersifat anorganik yang memiliki efek samping tubuh jika dikonsumsi dalam jangka waktu yang panjang (health.kompas post:2013). Alternatifnya adalah menggunakan tanaman-tanaman obat yang bersifat alami dan tidak menggunakan bahan-bahan sintetis.

Indonesia memiliki keanekaragaman jenis tanaman yang memiliki jumlah spesies 30.000 dan yang berpotensi menjadi obat sekitar 9.000 spesies tanaman (sriwijaya post:2012). Banyak manfaat yang dapat diperoleh dari keanekaragaman tanaman obat namun kurang diketahui oleh masyarakat dikarenakan kurangnya pendokumentasian. Ensiklopedia adalah salah satu sarana pendokumentasian keanekaragaman tanaman obat di Indonesia. Ensiklopedia tanaman obat tradisional adalah suatu buku yang dapat memberikan informasi kepada pengguna mengenai jenis tanaman apa saja yang dapat digunakan untuk menyembuhkan berbagai macam penyakit. Dengan ensiklopedia ini diharapkan pengguna dapat mengetahui informasi tanaman tersebut secara mendetail, mulai dari gambar tanaman tersebut, nama daerah, nama latin, dan cara mengolah tanaman tersebut sehingga dapat digunakan menjadi obat.

Pada masa ini kebutuhan manusia informasi sangatlah tinggi khususnya pencarian informasi keanekaragaman tanaman obat di Indonesia. Banyaknya informasi yang dibutuhkan oleh manusia maka bermunculan mesin-mesin

pencarian informasi. Mesin-mesin pencarian informasi yang telah beredar rata-rata memiliki bentuk sistem pencarian yang sama yaitu mengidentifikasi setiap satu kata. Pencarian satu kata ini akan menampilkan banyak dokumen yang memuat kata tersebut sehingga pencarian dokumen masih belum spesifik atau tidak sesuai dengan yang diharapkan. Pada pencarian berbasis *semantic web* ini menggunakan pencarian satu kalimat sehingga pencarian cenderung lebih akurat.

Pencarian informasi sebenarnya telah diriwayatkan dalam hadist Abu Dawud no.3641, bahwa Abud Darda' *radhiyallahu 'anhu* berkata: aku mendengar Rasulullah *shallahu 'alaihi wa salam* bersabda :

مَنْ سَلَكَ طَرِيقًا يَطْلُبُ فِيهِ عِلْمًا سَلَكَ اللَّهُ بِهِ طَرِيقًا مِنْ طُرُقِ الْجَنَّةِ وَإِنَّ الْمَلَائِكَةَ لَتَتَّصِعُ أَجْنِحَتَهَا رِضًا لِمَطَالِبِ الْعِلْمِ وَإِنَّ الْعَالِمَ لَيَسْتَغْفِرُ لَهُ مَنْ فِي السَّمَوَاتِ وَمَنْ فِي الْأَرْضِ وَالْحَيَاتَانُ فِي جَوْفِ الْمَاءِ وَإِنَّ فَضْلَ الْعَالِمِ عَلَى الْعَابِدِ كَفَضْلِ الْقَمَرِ لَيْلَةَ الْبَدْرِ عَلَى سَائِرِ الْكَوَاكِبِ وَإِنَّ الْعُلَمَاءَ وَرَثَةُ الْأَنْبِيَاءِ وَإِنَّ الْأَنْبِيَاءَ لَمْ يُورَثُوا دِينَارًا وَلَا دِرْهَمًا وَرَثُوا الْعِلْمَ فَمَنْ أَخَذَهُ أَخَذَ بِحِطَّةٍ وَافِرٍ

“Barangsiapa menempuh suatu jalan yang padanya dia mencari ilmu, maka Allah akan memudahkan dia menempuh jalan dari jalan-jalan (menuju) jannah, dan sesungguhnya para malaikat benar-benar akan meletakkan sayap-sayapnya untuk penuntut ilmu, dan sesungguhnya seorang penuntut ilmu akan dimintakan ampun untuknya oleh makhluk-makhluk Allah yang ada di langit dan yang ada di bumi, sampai ikan yang ada di tengah lautan pun memintakkan ampun untuknya. Dan sesungguhnya keutamaan seorang yang berilmu atas seorang yang ahli ibadah adalah seperti keutamaan bulan pada bulan purnama atas seluruh bintang, dan sesungguhnya ulama adalah pewaris para nabi dan para nabi tidaklah mewariskan dinar ataupun dirham, akan tetapi mereka hanya akan mewariskan ilmu, maka barangsiapa yang mengambilnya maka sesungguhnya dia telah mengambil bagian yang sangat banyak.”

Pada hadist yang diriwayatkan oleh Abu Dawud telah dijelaskan bahwa mencari ilmu akan diberikan kemuliaan yang sangat besar. Mencari informasi sama halnya dengan mencari ilmu, Allah memberikan banyak keistimewaan bagi orang-orang yang mau mencari informasi agar menambah ilmu pengetahuannya.

Mencari sebuah informasi sama halnya dengan mencari menggali pengetahuan untuk menambah ilmu seperti halnya yang telah diperintahkan oleh Allah SWT untuk terus mencari pengetahuan atau mencari informasi.

Perkembangan informasi yang semakin luas memicu pengembang teknologi membuat *mobile smartphone* yang bisa dibawa kemana-mana dan mudah dalam pencarian informasi. Munculnya ponsel pintar yang berkembang pesat ini menjadi sebuah gaya hidup masyarakat khususnya di Indonesia. Pada sebuah artikel di *viva news* menyebutkan bahwa *Smartphone Consumer Insight* melakukan riset pada Mei 2013 di Indonesia yang dilakukan oleh riset Global Nielsen menunjukkan rata-rata masyarakat Indonesia memanfaatkan *smartphone* selama 189 menit/hari atau setara dengan 3 jam 15 menit/hari dengan pengguna domain untuk *social media* dan *rich media*. Prosentase aktivitas paling tinggi yaitu *chatting* 90%, pencarian 71%, jejaring sosial 64%, *blogging/forum* 41 %, *app store* 32%, *video on demand* 27%, *sharing* konten 26%, hiburan 25%, berita 24% dan *webmail* 17%. Pencarian memiliki prosentase kedua tertinggi bahwa terbukti bahwa pencarian adalah aktivitas paling banyak pada pengguna *smartphone*. Pemanfaat teknologi *mobile smartphone* ini dapat digunakan untuk pencarian mengenai tanaman obat tradisional.

Pemanfaatan perkembangan teknologi ini maka akan dibuat aplikasi *mobile* ensiklopedia tanaman obat tradisional. Dengan menggunakan aplikasi ini, pengguna dapat mengetahui informasi tanaman langsung dari telepon seluler. Hal ini tentunya menjadi keunggulan tersendiri karena pengguna tidak usah repot membawa buku ensiklopedia yang berat ke mana-mana. Pengguna tinggal menggunakan aplikasi untuk mengetahui informasi tanaman obat tradisional tersebut.

1.2 Rumusan Masalah

Berdasarkan penjelasan pada latar belakang, perumusan masalah yang akan dikaji dalam penelitian ini yaitu seberapa akurat dan cepat sistem pencarian berbasis *semantic web* pada aplikasi ensiklopedia tanaman obat.

1.3 Tujuan Penelitian

Adapun tujuan yang dapat diambil dari penelitian ini yaitu membuat aplikasi mesin pencari tanaman obat tradisional berbasis *mobile* yang akurat dalam melakukan pencarian tanaman tradisional yang dapat digunakan sebagai obat.

1.4 Manfaat Penelitian

Adapun manfaat yang dapat diambil dari penelitian ini yaitu membantu masyarakat mendapatkan informasi dengan akurat tentang manfaat-manfaat tanaman yang ada dilingkungan sebagai obat herbal.

1.5 Batasan Masalah

Agar pembahasan dan penyusunan sesuai dengan tujuan yang diharapkan, maka perlu diberikan batasan-batasan masalah yaitu :

1. Sistem ini fokus terhadap pencarian menggunakan *ontology*.
2. Informasi mencakup tanaman-tanaman obat yang terdapat di Indonesia.
3. Spesifikasi hardware untuk menjalankan aplikasi *mobile* adalah 180 MB, 290 MB RAM.
4. Spesifikasi software untuk menjalankan aplikasi adalah OS Android 2.3.6 versi Gingerbread.
5. Pembuatan sistem berpusat pada penerapan *semantic search*.

1.6 Metode Penelitian

Peneliti membagi pengerjaan penelitian ini menjadi beberapa tahap, antara lain:

1. Studi literatur yaitu mengumpulkan data dan informasi tentang tanaman obat dan teknologi web semantik dari buku, artikel, jurnal dan berbagai literatur yang mendukung penyusunan laporan skripsi.
2. Studi eksperimen yaitu melakukan analisa dan perancangan sistem. Perancangan yang dilakukan ada dua yaitu perancangan *Semantic Search* dan perancangan aplikasi.
3. Implementasi fungsi pencarian semantik pada aplikasi *mobile*.
4. Melakukan uji coba pencarian tanaman obat dengan menggunakan kalimat.
5. Penyusunan laporan skripsi.

1.7 Sistematika Penulisan

Adapun sistematika pada skripsi ini adalah sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi tentang teori acuan yang berkaitan tentang permasalahan yang dibahas serta teori acuan dalam analisis dan pembuatan pemecahan permasalahan yang dibahas. Teori acuan tersebut yaitu, tanaman obat dan mesin pencari berbasis *semantic search* dan Algoritma *Boyer-Moore*

BAB III ANALISIS DAN PERANCANGAN

Bab ini berisi tentang analisis dan perancangan Mesin Pencari Berbasis *Semantic Search* Menggunakan Metode Algoritma *Booyer-Moore* Pada Ensiklopedia Tanaman Obat. Menganalisis kebutuhan dan tahap-tahap dalam pembuatan sistem.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi tentang implementasi dan hasil uji coba dari aplikasi Mesin Pencari Berbasis *Semantic Search* Menggunakan Metode Algoritma *Booyer-Moore* Pada Ensiklopedia Tanaman Obat.

BAB V PENUTUP

Bab ini berisi tentang kesimpulan dan saran yang diharapkan dapat bermanfaat dalam pengembangan program aplikasi.



BAB II

TINJAUAN PUSTAKA

2.1 Esiklopedia

Ensiklopedi atau “Encyclopaedia” berasal dari bahasa Yunani *enkyklios paideia* (ἐγκύκλιος παιδεία). *Enkyklios* yang berarti siklus atau perputaran dan *paedia* berarti pengetahuan atau widya (Anonim, 2011). Secara sederhana pengertian Encyclopaedia adalah sebuah lingkaran atau siklus ilmu pengetahuan.

Dalam Kamus Besar Bahasa Indonesia, definisi ensiklopedi adalah buku yang berisi keterangan atau uraian ringkas tentang berbagai-bagai hal dalam ilmu pengetahuan yang disusun menurut abjad atau menurut lingkungan ilmu (Poerwadarminta, 1993).

Ensiklopedi digolongkan kedalam kelompok buku referensi bersama kamus, *companion*, *consice*. Buku referensi sering digunakan sebagai kajian, rujukan untuk mendapatkan informasi dalam suatu bidang keilmuan. Ensiklopedi berbeda dengan kamus. Kamus berfungsi untuk mendefinisikan atau menerangkan arti kata yang terkandung dan keterkaitan dengan kebahasaan, sedangkan ensiklopedi mendefinisikan secara lebih luas yang mencakup artikel, topik, uraian, definisi, dan rangkaian artikel dari awal sampai akhir serta peristiwa-peristiwa penting dalam ilmu pengetahuan.

Secara umum, ensiklopedi terdiri atas dua kategori yaitu ensiklopedi umum dan ensiklopedi khusus. Ensiklopedi umum berisi berbagai informasi dari semua cabang ilmu yang di dalamnya dibahas secara umum. Contoh ensiklopedi umum yaitu “Ensiklopedi Nasional Indonesia oleh PT Delta Pamungkas 2004”.

Sedangkan ensiklopedi khusus hanya berisi satu cabang atau bidang keilmuan tertentu. Contoh ensiklopedi khususnya yaitu “Ensiklopedi Etika Islam oleh Abdul Aziz bin Fathi as-Syaid Nada 2006”.

Seiring perkembangan informasi digital yang berkembang pesat. Ensiklopedi yang awalnya berupa buku tebal dan hanya bisa didapatkan di perpustakaan, kini ensiklopedia sudah berubah dalam bentuk perangkat lunak maupun *website*. Banyak situs web yang menyediakan ensiklopedi secara *online* yang dapat diakses dengan mudah dimanapun berada melalui jaringan internet. Contoh Wikipedia dan *Britannica* (Anonim, 2011).

2.2 Tanaman Obat

Bagian tumbuhan yang sering dimanfaatkan untuk pengobatan adalah bagian kormus. Kormus terdapat pada divisi *Pteridophyta* (tumbuhan paku) dan *Spermatophyta* (tumbuhan biji) (Savitri, 2008).

Tumbuhan kormus (*Cormophyta*) adalah tumbuhan yang memiliki tiga bagian pokok tumbuhan yaitu akar (*radix*), batang (*caulis*), dan daun (*folium*). Tumbuhan kormus medapat dimanfaatkan sebagai obat-obatan dikarenakan pada bagian daun, batang, akar, rimpang, bunga, buah, dan bijinya mengandung zat kimia yang dapat dimanfaatkan untuk mengobati penyakit maupun untuk menjaga daya tahan tubuh terhadap penyakit. (Savitri, 2008).

Allah SWT menciptakan manusia di muka bumi agar manusia menjadi khalifah seperti yang disebutkan dalam Surat Al-Baqarah ayat 30.

“Ingatlah ketika Tuhanmu berfirman kepada para Malaikat: “Sesungguhnya Aku hendak menjadikan seorang khalifah di muka bumi.” Mereka berkata: “Mengapa Engkau hendak menjadikan (khalifah) di bumi itu orang yang

akan membuat kerusakan padanya dan menumpahkan darah, padahal kami senantiasa bertasbih dengan memuji Engkau dan mensucikan Engkau?” Tuhan berfirman: “Sesungguhnya Aku mengetahui apa yang tidak kamu ketahui.”

Khalifah yang disebutkan pada ayat diatas yakni sebagai penguasa yang mengatur serta memanfaatkan untuk kemaslahatan. Allah SWT menciptakan tumbuhan yang bermacam-macam jenisnya yang dapat dimanfaatkan sebagai obat dari berbagai penyakit, dan ini merupakan anugrah dari Allah SWT yang harus dipelajari dan dimanfaatkan seperti disebutkan dalam Quran Surat Al-Qashash ayat 57

“ dan mereka berkata : jika kami mengikuti petunjuk bersama kamu, niscaya kami akan diusir dari negeri kami.” Dan apakah kami tidak meneguhkan kedudukan mereka dalam daerah haram (tanah suci) yang aman, yang didatangkan ke tempat itu buah-buahan dari segala macam (tumbuh-tumbuhan) untuk menjadi rezeki (bagimu) dari sisi Kami ? tetapi kebanyakan mereka tidak mengetahui.”

Surat Al-Qashash ayat 57 mengisyaratkan agar kita mencari dan mempelajari bebrbagai tumbuhan yang menjadi rezeki yaitu yang memberi manfaat bagi kehidupan. Tumbuhan menjadi rezeki bagi makhluk hidup karena merupakan bahan pangan, bahan sandang, papan dan bahan obat-obatan. (Savitri, 2008).

2.2.1 Bagian-bagian Tumbuhan

Berikut akan diuraikan bagian-bagian tumbuhan satu persatu, baik bagian-bagian yang tergolong dalam alat hara maupun bagian-bagian yang tergolong alat perkembangbiakan :

2.2.1.1 Alat Hara

Semua bagian tubuh tumbuhan yang secara langsung ataupun tidak langsung berguna untuk menegakkan kehidupan tumbuhan, yaitu yang terutama berguna untuk penyerapan, pengolahan, pengangkutan dan penimbunan zat-zat makanan dinamakan alat hara.

2.2.1.2 Daun

Daun merupakan suatu bagian tumbuhan yang penting dan pada umumnya tiap tumbuhan mempunyai sejumlah besar daun. Daun biasanya tipis melebar, kaya akan suatu zat warna hijau yang dinamakan klorofil. Oleh karena itu daun biasanya berwarna hijau dan menyebabkan tumbuhan atau daerah-daerah yang ditempati tubuh-tumbuhan nampak hijau pula.

2.2.1.3 Batang

Batang merupakan bagian tubuh-tumbuhan yang amat penting dan mengingatkan tempat serta kedudukan batang bagi tubuh tumbuhan, batang dapat disamakan dengan sumbu tubuh tumbuhan.

2.2.1.4 Akar

Akar adalah bagian pokok ke-tiga selain batang dan daun bagi tumbuhan yang tubuhnya telah merupakan kormus.

2.2.1.5 Bagian-Bagian Lain

Diantara berbagai macam bagian tumbuhan yang sering dijumpai, yang tidak jelas berupa akar, batang atau daun adalah:

- **Kuncup**, merupakan bagian tumbuhan yang sesungguhnya adalah calon tunas.
- **Rimpang**, umbi (*tuber*) dan umbi lapis (*bulbus*), ketiga macam alat tersebut adalah metamorfosis dari batang dan akar atau daun.
- **Alat Pobelit**, yang dinamakan alat pobelit adalah bagian tumbuhan yang biasanya menyerupai spiral dan berguna pobelit benda-benda yang disentuhnya.
- **Piala dan gelebung**, piala biasanya merupakan ujung daun yang menjadi badan menyerupai piala yang lengkap dengan tutupnya. Gelebung terdapat pada tumbuhan pemakan serangga yang hidup di air.
- **Duri**, duri hanya sebagai alat tambahan saja.
- **Alat-alat tambahan**, alat tambahan dibedakan menjadi 3 golongan yaitu pappila, trikoma dan emergensia.

2.2.1.6 Alat Perkembangbiakan

Alat perkembangbiakan adalah bagian tumbuhan yang kemudian dapat tumbuh menjadi individu baru.

2.2.1.7 Bunga

Alat perkembangbiakan generatif itu bentuk dan susunannya berbeda-beda menurut jenisnya, tetapi bagi tumbuhan berbiji, alat tersebut lazimnya merupakan bagian tumbuhan yang dikenal sebagai bunga.

2.2.1.8 Buah

Peristiwa penyerbukan yang telah terjadi kemudian diikuti pula oleh pembuahan, maka bakal buah akan tumbuh menjadi buah dan bakal biji yang terjadi di dalam bakal buah akan tumbuh menjadi biji.

2.2.1.9 Biji

Biji merupakan alat perkembangbiakan yang utama, karena biji mengandung calon tumbuhan baru (lembaga). (Savitri, 2008).

2.2.2 Tumbuhan yang berkhasiat

Beberapa jenis tumbuhan obat yang berkhasiat menurut perspektif islam yaitu :

2.2.2.1 Semangka

Semangka mampu menumpas bibit kanker. Banyak memakan semangka dapat menyusutkan risiko kanker mulut rahim dan kanker pankreas pada wanita yang beresiko tinggi yang sehari-hari dietnya sarat kalori, gula, lemak tapi miskin sayur dan buah.

2.2.2.2 Bawang Merah

Bawang merah dapat digunakan untuk mencegah kanker perut. Mengonsumsi bawang meah setiap hari bisa membantu pertumbuhan jaringan tulang dan mengurangi resiko osteoporosis.

2.2.2.3 Terong

Terong dapat digunakan sebagai penurun tekanan darah tinggi.

2.2.2.4 Jintan Hitam

Jintan hitam digunakan sebagai peningkatan sistem kekebalan tubuh manusia.

2.2.2.5 Anggur

Anggur terbukti dapat membantu perempuan mencegah sakit akibat pengobatan kanker payudara. (Savitri, 2008).

2.3 *Semantic*

2.3.1 *Semantic Web*

Dalam sejarahnya, web dikembangkan oleh Tim Berners-Lee pada tahun 1990 dengan memperkenalkan Hypertext Markup Language (HTML) sebagai standar untuk mengatur system dokumen yang terdistribusi. Web dibangun untuk memenuhi kebutuhan manusia, sehingga web dirancang dengan sifat *machine readable*, dan bukan *machine understandable* (Berners dkk, 2001). Untuk menghasilkan halaman web yang terkesan dinamis maka HTML dalam perkembangannya banyak menggunakan basis data. Tetapi dengan bertambahnya kebutuhan, maka metode ini mulai dirasakan kelemahannya. Dibutuhkan format data yang dapat memisahkan antara deskripsi isi (content) dan deskripsi presentasi (*layout*). *eXtensible Markup Language* (XML) adalah bahasa pemrograman yang dikembangkan untuk membangun sebuah aplikasi dengan fungsional lebih dibandingkan HTML. Tidak seperti HTML, XML tidak mendefinisikan bagaimana data tersebut dapat disajikan, namun XML mengatur proses penyajian data sehingga dapat ditampilkan secara lebih fleksibel. XML

memberikan deskripsi isi data, sehingga perangkat lunak dapat memahami konteks data secara otomatis. (Siallagan, 2009).

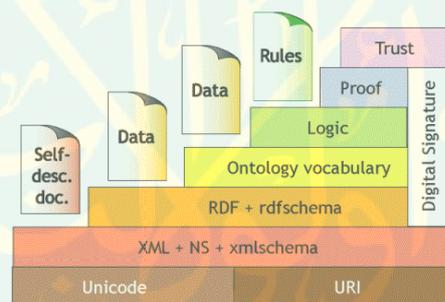
Saat ini, web terdiri atas banyak sekali informasi dalam bentuk website yang dirancang untuk penggunaan secara manual oleh manusia. Tidak seperti mesin, seorang pembaca dengan mudah dapat mengambil informasi dari sebuah website dan mengaitkannya dengan informasi lain. Akibatnya sulit menyaring informasi tertentu dari sekian banyak informasi yang tersedia. Search engine seperti google, selama ini memberikan hasil yang diminta oleh pengguna berdasarkan metode statistic, dengan algoritma pagerank.

Kata semantic berarti makna atau sesuatu yang berhubungan dengan ilmu yang mempelajari makna dan perubahan makna. Dalam konteks *semantic web* kata *semantic* menunjukkan bahwa makna suatu data yang terdapat di dalam web dapat dipahami bukan hanya oleh manusia namun juga oleh mesin (*machine understandable*). Tim Berners-Lee mengatakan *semantic web* merupakan upaya untuk menyamakan persepsi antar konsep dari definisi sebuah informasi sehingga kerjasama antara komputer dan manusia dalam pengolahan informasi bisa lebih optimal (Berners, 2001).

Menurut Word Wide Web Consortium (W3C) *semantic web* adalah sebuah gambaran dari sebuah gagasan agar data yang terdapat didalam web dapat didefinisikan dan direlasikan dalam suatu cara sehingga dapat digunakan oleh mesin, bukan hanya sekedar untuk ditampilkan, tetapi juga untuk diotomatisasi, diintegrasikan, dan digunakan kembali dalam aplikasi yang berbeda. Istilah *semantic web* berawal dari gagasan untuk menyediakan data di internet

sedemikian rupa agar proses logis dengan mesin dapat dilakukan. Untuk itu, pada semantic web, selain informasi murni, terdapat pula keterangan mengenai keterkaitan informasinya. Agar dapat direalisasikan W3C menetapkan beberapa *open standard*. Peranan penting dalam hal ini ada pada bahasa komputer XML, RDF, OWL, dan SPARQL (*Query Language RDF*). Bahasa-bahasa tersebut memungkinkan penyajian informasi secara semantic dalam bentuk *ontology* dan *taksonomi* (W3C, 2012).

Arsitektur lapisan dalam pembangunan semantic web adalah sebagai berikut:



Gambar 2.1. Arsitektur *Semantic Web* (sumber: Tim Berners-Lee)

Semantic Web mempunyai beberapa standar operasional untuk menjalankan fungsinya. Dalam menampung metadata *semantic web* menggunakan Resource Description Framework (RDF) dan Web Ontology Language (OWL). Dalam teknologi *semantic web* peran RDF (Resource Description Framework) berfungsi untuk mendefinisikan format metadata yang terdiri dari *subject*, *predicate*, dan *object*. *Subject* dan *object* adalah entitas yang ditunjukkan oleh teks (Media Iptek, 2006). Sedangkan predicate adalah komposisi yang menerangkan sudut pandang dari subject yang dijelaskan object. Didalam format RDF suatu *object* dapat menjadi *subject* yang nantinya diterangkan oleh *object* yang lainnya.

Sehingga *object* dapat diterangkan secara jelas dan detail, serta sesuai dengan keinginan pengguna yang memberikan masukan. *semantic web* merujuk kepada kemampuan aplikasi komputer untuk lebih memahami bahasa manusia, dimana komputer dapat memahami bahasa yang kompleks dari para penggunanya, seperti dalam bahasa percakapan sehingga memudahkan penggunaan dalam berkomunikasi dengan mesin. *Semantic web* dapat mengolah bahasa dan mengenali homonim, sinonim, atau atribut yang berbeda.

Standar dalam membangun *Semantic Web* adalah sebagai:

1. XML (*eXtended Markup Language*)

XML (*eXtended Markup Language*) adalah bahasa markup yang digunakan untuk menyimpan. Markup yaitu bahasa yang berisikan kode-kode berupa tanda-tanda tertentu dengan aturan tertentu untuk memformat dokumen teks dengan menggunakan tag yang dibuat sendiri.

2. XML Schema

XML Schema adalah alternatif berbasis XML untuk DTD. Sebuah skema XML berfungsi untuk menjelaskan struktur dari dokumen XML, *XML Schema* juga disebut sebagai XML Schema Definition (XSD). Tujuan dari XML Schema adalah untuk mendefinisikan blok dari dokumen XML, seperti sebuah DTD.

Fungsi *XML Schema* adalah sebagai berikut:

- Mendefinisikan unsur-unsur yang dapat muncul dalam dokumen.
- Mendefinisikan atribut yang dapat muncul dalam dokumen.
- Mendefinisikan elemen elemen turunan.
- Mendefinisikan urutan elemen turunan.

- Mendefinisikan jumlah elemen turunan
- Mendefinisikan *value* dari suatu elemen.
- Mendefinisikan tipe data untuk elemen dan atribut
- mendefinisikan nilai default dan tetap untuk elemen dan atribut

3. RDF (*Resource Description Framework*).

RDF atau *Resource Description Framework* merupakan model standar untuk pertukaran data pada web. Menurut Liang Yu pada bukunya yang berjudul *A Developer's Guide to the Semantic Web*, RDF merupakan sebuah bahasa untuk mendeskripsikan informasi tentang resource web. Dalam RDF informasi dideskripsikan menjadi sebuah kalimat yang terdiri dari subjek, predikat dan objek. Pada RDF resource yang digunakan untuk mendeskripsikan subjek biasanya menggunakan alamat URI, predikat merupakan property yang menghubungkan antara subjek dan objek dan objek adalah URI atau literal. Ketiga elemen tersebut dikenal dengan sebutan *triple* (Liyang, 2007).

4. RDF Schema

RDF Schema (*Resource Description Framework Schema*) yang biasa dikenal dengan berbagai singkatan yaitu: RDFS , RDF (S), RDF-S, atau RDF / S. RDF Schema merupakan skema yang mendeskripsikan ontologi yang berupa *properties* dan *class* dari RDF resource.

5. OWL

Web Ontology Language (OWL) adalah suatu bahasa yang digunakan oleh aplikasi untuk pemrosesan isi dari informasi tersebut. Dengan menggunakan

OWL dapat ditambahkan *vocabulary* tambahan selain *semantic formal* yang telah dibuat menggunakan RDF, RDFS, XML dan XML *Schema*.

Perancangan OWL dapat dibuat dalam 3 buah subbahasa yaitu :

- OWL Lite dalam pembuatannya membutuhkan suatu hirarki pengklasifikasian dan berbagai constraints sederhana.
- OWL DL dapat digunakan untuk membuat *ontology* dengan tingkat ekspresi maksimal dan semua konklusi yang dihasilkan dapat dihitung dalam waktu yang terbatas.
- OWL full, dapat digunakan untuk membuat *ontology* dengan tingkat ekspresi maksimal dan kebebasan sintaks dari RDF tanpa mempertimbangkan komputasi yang dibutuhkan.

2.3.2 *Semantic Search*

Semantic search adalah sebuah teknologi yang bertujuan untuk meningkatkan akurasi dalam melakukan pencarian dengan cara memahami dan makna kontekstual dari istilah yang dicari, sehingga mendapatkan hasil yang lebih relevan. (Pollock, 2009).

Sistem *semantic search* melakukan pencarian dengan mempertimbangkan beberapa kondisi yaitu: konteks pencarian, variasi, makna kata, sinonim, konsep umum-khusus, konsep SPO(subjek, predikat, objek) dan pernyataan bahasa alami untuk menyediakan hasil pencarian yang relevan. (Pollock, 2009).

2.4 Ontologi

Ontologi dalam bidang kecerdasan buatan memiliki dua pengertian yang berkaitan. Pertama ontologi merupakan kosakata yang merepresentasikan suatu domain atau subyek tertentu. Kedua ontologi merupakan *body of knowledge* untuk menjelaskan suatu bahasan tertentu. Dari kedua hal tersebut dapat diambil kesimpulan bahwa ontology merupakan sebuah uraian formal yang merepresentasikan sebuah konsep atau *class* dari sebuah domain tertentu. Properti-properti dari setiap konsep atau *class* dijelaskan didalam atribut-atribut.

Sebuah ontologi bersama dengan beberapa set *instances* dari *class* akan membentuk sebuah presentasi pengetahuan. Secara umum, ontologi digunakan pada kecerdasan buatan untuk merepresentasikan sebuah pengetahuan.

OWL (*Ontology Web Language*) merupakan bahasa ontologi yang direkomendasikan oleh W3C. OWL digunakan untuk mempresentasikan arti dari istilah-istilah suatu objek pengetahuan baik secara eksplisit maupun relasi. OWL memiliki fasilitas yang lebih baik dibandingkan XML, RDF/S dalam mendeskripsikan arti. (McGuinness, 2004).

2.5 Platform Android

Android adalah sistem operasi berbasis linux yang dipergunakan untuk perangkat *mobile* seperti *smartphone* dan komputer tablet. Android menyediakan platform yang bersifat terbuka (*open source*) sehingga bagi para *developer* dapat membuat aplikasi mereka sendiri untuk digunakan pada perangkat *mobile*.

Arsitektur perancangan pada *platform* Android adalah sebagai berikut:

- *Application* dan *Widgets* merupakan layer untuk menghubungkan pengguna dengan aplikasi, pada layer ini dapat ditemukan fungsi-fungsi dasar *smartphone* seperti untuk sms dan telepon.
- *Application Frameworks* merupakan layer untuk para developer dalam membangun atau mengembangkan aplikasi yang akan dijalankan pada sistem operasi Android.
- *Libraries* merupakan layer untuk tempat fitur-fitur pada Android, fitur-fitur tersebut digunakan untuk menjalankan aplikasi yang dibangun pada sistem operasi Android.
- *Android Run Time* merupakan layer yang berisi *Core Libraries* dan *Dalvik Virtual Machine (DVK)*. *Core libraries* mencakup serangkaian inti *library* Java yang menyediakan sebagian besar fungsi-fungsi yang ada pada *library* dasar bahasa pemrograman Java. Sedangkan DVK merupakan *register bases* untuk menjalankan Android.

Android menggunakan kernel linux sebagai inti dari sistem operasinya, dengan mengutamakan stabilitas dan ketersediaan driver, linux sebagai kernel monolitik menyediakan semua yang dibutuhkan untuk menjalankan seluruh sistem. Google menggunakan kernel Linux versi 2.6 untuk membangun sistem Android, yang mencakup *memory management*, *security setting*, *power management* dan beberapa *driver hardware*. Kernel berperan sebagai *abstraction layer* antara hardware dan keseluruhan software. (Nazruddin, 2011).

2.6 Algoritma Boyer-Moore

Algoritma Boyer-Moore adalah algoritma pencarian *string* yang melakukan pencocokan karakter dari sebelah kanan *pattern*. Algoritma Boyer-Moore dibuat oleh R.M Boyer dan J.S Moore.

Proses pencocokan *string algoritma* ini adalah dengan cara membandingkan karakter paling kanan dari *pattern* yang dicari. Dengan proses pencocokan yang dimulai dari sebelah kanan proses pencarian akan menjadi lebih cepat dikarenakan biasanya informasi terpenting dari *string* berada pada sebelah kanan. Proses pencocokan dari kanan (posisi terakhir *pattern* yang dicari) ditunjukkan dalam gambar 2.2 :

<i>m</i>	<i>a</i>	<i>k</i>	<i>a</i>	<i>n</i>		<i>j</i>	<i>a</i>	<i>m</i>	<i>b</i>	<i>u</i>
<i>j</i>	<i>a</i>	<i>m</i>	<i>b</i>	<i>u</i>						

Gambar 2.2. *Pattern Boyer-Moore* (Edward G.R., Algoritma boyer-moore)

Pada gambar 2.2, proses pencocokan *string* dimulai dari *pattern* paling kanan yaitu karakter “u” pada *string* “jambu” tidak cocok dengan karakter “n” pada *string* “makan” dari text yang dicari, dan karakter “n” tidak ada dalam *string* “jambu” maka dilakukan pergeseran pada *pattern* “jambu” sehingga posisinya menjadi seperti pada gambar 2.3.

<i>m</i>	<i>a</i>	<i>k</i>	<i>a</i>	<i>n</i>		<i>j</i>	<i>a</i>	<i>m</i>	<i>b</i>	<i>u</i>
					<i>j</i>	<i>a</i>	<i>m</i>	<i>b</i>	<i>u</i>	

Gambar 2.3. *Pergeseran Pattern Boyer-Moore* (Edward G.R., Algoritma boyer-moore)

Dari gambar 2.3 pergeseran dilakukan sebanyak 5 karakter dikarenakan tidak ada karakter yang cocok sehingga mempercepat proses pencarian (Christabella, 2011).

Langkah-langkah algoritma Boyer-Moore :

- Buat tabel pergeseran string yang dicari (S) dengan pendekatan Match Heuristic (MH) dan Occurrence Heuristic (OH), untuk menentukan jumlah pergeseran yang akan dilakukan jika mendapat karakter tidak cocok pada proses pencocokan dengan string (T).
- Jika dalam proses perbandingan terjadi ketidakcocokan antara pasangan karakter pada S dan karakter pada T, pergeseran dilakukan dengan memilih salah satu nilai pergeseran dari dua tabel analisa string, yang memiliki nilai pergeseran paling besar.
- Dua kemungkinan penyelesaian dalam melakukan pergeseran S, jika sebelumnya belum ada karakter yang cocok adalah dengan melihat nilai pergeseran hanya pada tabel occurrence heuristic : Jika karakter yang tidak cocok tidak ada pada S maka pergeseran adalah sebanyak jumlah karakter pada S. dan jika karakter yang tidak cocok ada pada S, maka banyaknya pergeseran bergantung dari nilai pada tabel.
- Jika karakter pada teks yang sedang dibandingkan cocok dengan karakter pada S, maka posisi karakter pada S dan T diturunkan sebanyak 1 posisi, kemudian lanjutkan dengan pencocokan pada posisi tersebut dan seterusnya. Jika kemudian terjadi ketidakcocokan karakter S dan T, maka pilih nilai pergeseran terbesar dari dua tabel analisa

pattern yaitu nilai dari tabel match heuristic dan nilai tabel occurrence heuristic dikurangi dengan jumlah karakter yang telah cocok.

- Jika semua karakter telah cocok, artinya S telah ditemukan di dalam T, selanjutnya geser pattern sebesar 1 karakter.
- Lanjutkan sampai akhir string T.

Cara Menghitung Tabel *Occurence Heuristic* :

Tabel 2.1: *Occurence Heuristic*

posisi:	1	2	3	4	5	6	7
string:	m	a	n	a	m	a	n
Pergeseran(OH):	2	1	0	1	2	1	0

- Lakukan pencacahan mulai dari posisi terakhir string sampai ke posisi awal, dimulai dengan nilai 1, catat karakter yang sudah ditemukan (dalam contoh ini karakter “n”).
- Mundur ke posisi sebelumnya, nilai pencacah ditambah 1, jika karakter pada posisi ini belum pernah ditemukan, maka nilai pergeserannya adalah sama dengan nilai pencacah. (dalam contoh ini, karakter “a” belum pernah ditemukan sehingga nilai pergeserannya adalah sebesar nilai pencacah yaitu 2)
- Mundur ke posisi sebelumnya, karakter “m” nilai pergeserannya 3
- Mundur lagi, karakter “a”. karakter “a” sudah pernah ditemukan sebelumnya sehingga nilai pergeserannya sama dengan nilai pergeseran karakter “a” yang sudah ditemukan paling awal yaitu 2.
- Begitu seterusnya sampai posisi awal string.

Cara Menghitung Tabel *Match Heuristic* :

Tabel 2.2: Match Heuristic

posisi:	1	2	3	4	5	6	7
string:	m	a	n	a	m	a	n
Pergeseran(MH):	4	4	4	4	7	7	1

Langkah-langkah perhitungannya adalah sebagai berikut :

- Jika karakter pada posisi 7 bukan “n” maka geser 1 posisi, berlaku untuk semua string yang dicari.

Tabel 2.3: Pergeseran pertama

teks:	m	a	n	a	m	a	b	m	a	n	a	m	a	n
string	m	a	n	a	m	a	n							
		m	a	n	a	m	a	n						

- Jika karakter “n” sudah cocok, tetapi karakter sebelum “n” bukan “a” maka geser sebanyak 7 posisi, sehingga posisi string melewati teks.karena sudah pasti “manambn” bukan “manaman”.

Tabel 2.4: Pergeseran kedua

teks:	m	a	n	a	m	b	n	m	a	n	a	m	a	n
string	m	a	n	a	m	a	n							
								m	a	n	a	m	a	n

- Jika karakter “an” sudah cocok, tetapi karakter sebelum “an” bukan “m” maka geser sebanyak 7 posisi, sehingga posisi string melewati teks.karena sudah pasti “manaban” bukan “manaman”.

Tabel 2.5: Pergeseran ketiga

teks:	m	a	n	a	b	a	n	m	a	n	a	m	a	n
string	m	a	n	a	m	a	n							
								m	a	n	a	m	a	n

- Jika karakter “man” sudah cocok, tetapi karakter sebelum “man” bukan “a” maka geser sebanyak 4 posisi, sehingga posisi string berada / bersesuaian dengan akhiran “man” yang sudah ditemukan sebelumnya. karena bisa saja akhiran “man” yang sudah ditemukan

sebelumnya merupakan awalan dari string “manaman” yang berikutnya.

Tabel 2.6: Pergeseran keempat

teks:	m	a	n	b	m	a	n	m	a	n	a	m	a	n
string	m	a	n	a	m	a	n							
								m	a	n	a	m	a	n

- Jika karakter “aman” sudah cocok, tetapi karakter sebelum “aman” bukan “n” maka geser sebanyak 4 posisi, sehingga posisi string berada / bersesuaian dengan akhiran “man” yang sudah ditemukan sebelumnya, karena bisa saja akhiran “man” yang sudah ditemukan sebelumnya merupakan awalan dari string “manaman” yang berikutnya.

Tabel 2.7: Pergeseran kelima

teks:	m	a	b	a	m	a	n	m	a	n	a	m	a	n
string	m	a	n	a	m	a	n							
								m	a	n	a	m	a	n

- Selanjutnya sama, pergeseran paling mungkin dan aman dalam tabel Match Heuristic adalah pergeseran sebanyak 4 posisi. (Edward, 2009)

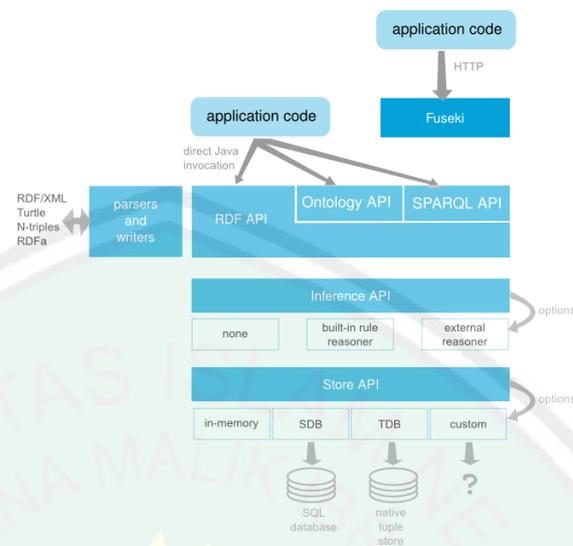
2.7 Jena Framework

Jena adalah *framework* dalam bahasa pemrograman java yang digunakan untuk membangun aplikasi *Semantic Web*. Jena menyediakan *library* Java yang luas untuk membantu *developers* mengembangkan *source code* yang dapat menangani RDF, RDFS, RDFa, OWL dan SPARQL. Jena mencakup mesin berbasis aturan inferensi untuk melakukan penalaran berdasarkan OWL, RDFS ontologi dan berbagai strategi penyimpanan baik dalam *n-triple*, memori maupun pada disk (<http://jena.apache.org> : 2012)

Kerangka Jena meliputi:

- API untuk membaca, pengolahan dan menulis RDF data dalam XML, *N-Triple* dan *Turtle Formats*;
- API ontologi untuk menangani OWL dan RDFS ontologi;
- Mesin inferensi berbasis aturan untuk penalaran dengan RDF dan sumber data OWL;
- *Store* memungkinkan sejumlah besar RDF *triple* akan efisien disimpan pada disk;
- Mesin sesuai permintaan dengan terbaru SPARQL spesifikasi
- Server memungkinkan data RDF yang akan dipublikasikan ke aplikasi lain yang menggunakan berbagai protokol, termasuk SPARQL (<http://jena.apache.org> : 2012).

Pada intinya, Jena menyimpan informasi sebagai RDF *triple* dalam grafik diarahkan, dan memungkinkan menambahkan kode, menghapus, memanipulasi, menyimpan dan mempublikasikan informasi tersebut. Rata-rata berpikir Jena sebagai sejumlah subsistem utama dengan antarmuka yang jelas antara mereka. Berikut ini adalah gambaran besar dari jena:



Gambar 2.4. Arsitektur Jena (sumber: Aphace Jena)

2.8 Protégé

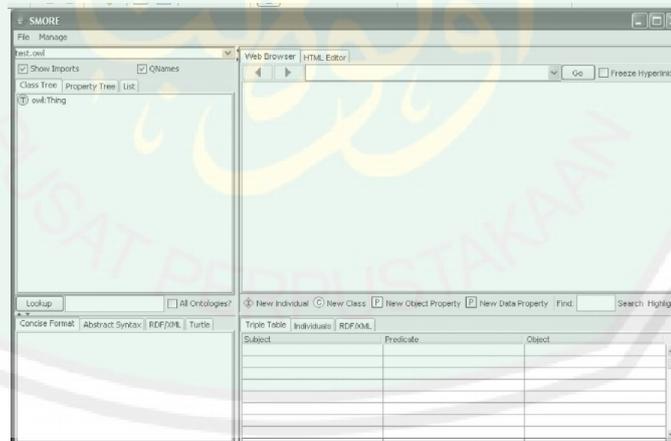
Protégé adalah alat yang digunakan untuk membuat sebuah domain ontologi, menyesuaikan form untuk entry data, dan memasukkan data. Format penyimpanannya antara lain OWL, RDF, XML, dan HTML. Alat ini memberikan kemudahan *plug and play* sehingga sangat fleksibel untuk pengembangan *prototype*. Protégé dibangun menggunakan bahasa pemrograman Java dan alat-alatnya dapat digunakan melalui *Graphical User Interface* (GUI) dalam bentuk Tab bagi masing-masing bagian dan fungsi. Untuk mendefinisikan *class*, hirarki *class*, properti, nilai properti, relasi antar *class*, dan properti antar relasi tersebut, pengguna menggunakan *class* Tab (Horridge, 2011).

Protégé merupakan sebuah alat yang digunakan untuk membuat sebuah domain *ontology*, menyesuaikan form entry data, dan memasukkan data. Berbagai format penyimpanannya seperti OWL, RDF, XML, dan HTML. *Protégé* menyediakan kemudahan *plug and play* yang

membuatnya fleksibel untuk pengembangan *prototype*. *Protégé* dibuat dengan bahasa pemrograman Java. Semua alat-alat dalam *protégé* dapat digunakan melalui *Graphical User Interface* (GUI) dengan menyediakan Tab untuk masing-masing bagian dan fungsi standar.

2.9 Smore

Smore merupakan aplikasi untuk melakukan *markup document* yang dikembangkan oleh para peneliti di Universitas Maryland di College Park menggunakan bahasa pemrograman java. Smore digunakan untuk melakukan markup dokumen web dari file bertipe ontologi OWL. Smore menggunakan *Graphical User Interface* (GUI) yang memudahkan pengguna untuk membentuk ontologi serta dapat digunakan sebagai *ontology validator* (Liyang, 2007).

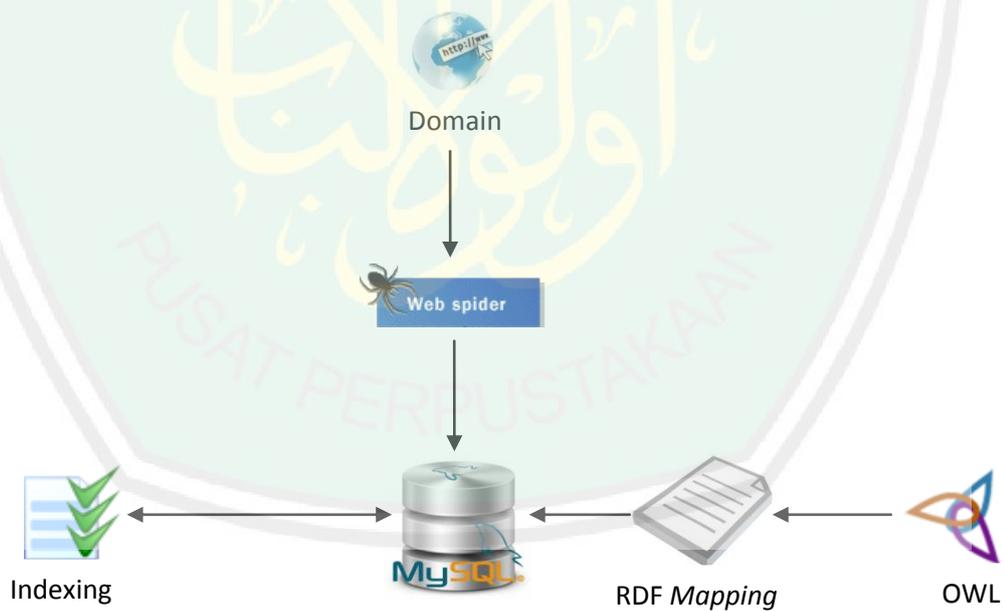


Gambar 2.5. Smore Web Page (sumber: Liyang)

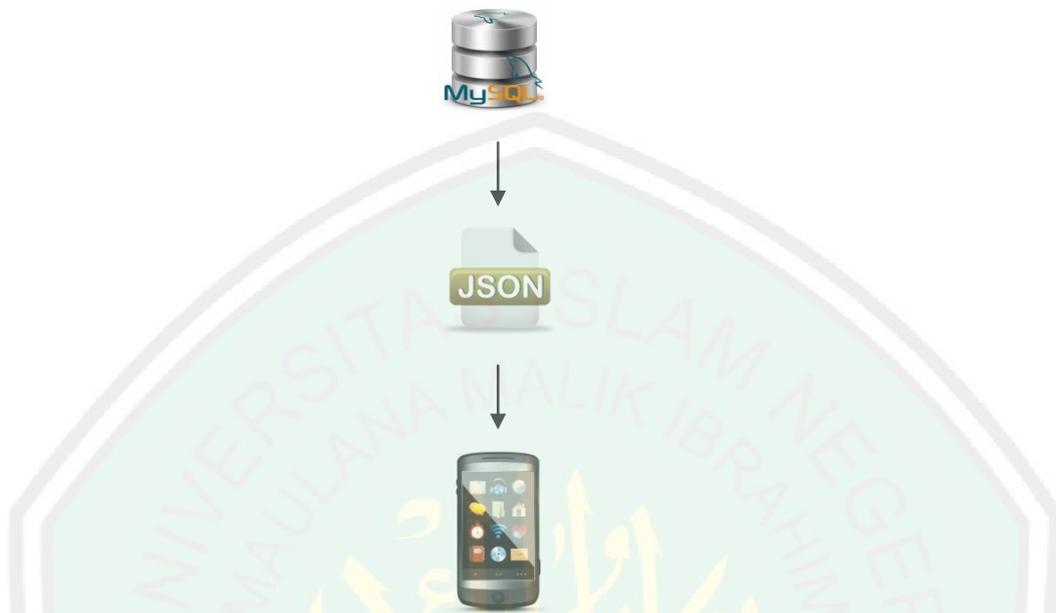
BAB III

ANALISIS DAN PERANCANGAN

Pada penelitian ini terdapat dua tahap dalam pembuatan aplikasi ini yaitu tahap *preprocessing* dan *processing*. Pada tahap pertama yaitu tahap *preprocessing* atau menyiapkan data-data yang dibutuhkan. Pada tahapan *preprocessing* terdapat proses *crowling*, *indexing* dan *RDF Mapping*. Pada tahap kedua yaitu *processing*. Pada tahap ini adalah proses pencarian yang dilakukan langsung dari *smartphone* ke server.



Gambar 3.1 Rancangan aplikasi tahap *preprocessing*



Gambar 3.2 Rancangan aplikasi tahap *processing*

3.1 Sumber Data Tanaman

Tanaman yang tumbuh di Indonesia terdapat tanaman hias, tanaman beracun, tanaman konsumsi dan tanaman berkhasiat. Tanaman di dunia tumbuh sesuai dengan manfaatnya masing-masing. Pada penelitian ini akan dibahas mengenai tanaman obat atau tanaman berkhasiat.

Tanaman dalam penelitian ini adalah tanaman yang bermanfaat bagi makhluk hidup. Pada penelitian ini adalah tanaman yang dibudidayakan dan ditanam yang dikelompokkan menjadi suatu tanaman memiliki manfaat sebagai obat yang berguna untuk dikonsumsi sebagai obat yang alami serta dapat menyembuhkan berbagai penyakit.

Data tanaman khasiat yang digunakan pada penelitian ini diambil dari buku Atlas Tumbuhan Indonesia Jilid 3. Penjelasan pada buku dijabarkan secara sistematis. Setiap tumbuhan dijelaskan dengan format sebagai berikut :

1. Nama Indonesia
2. Nama Latin
3. Suku
4. Sinonim
5. Nama Daerah
6. Nama Asing
7. Nama Simplisia
8. Uraian Tumbuhan (menjelaskan ciri-ciri)
9. Sifat dan Khasiat
10. Kandungan Kimia
11. Bagian yang Digunakan
12. Indikasi
13. Cara Pemakaian
14. Efek Farmakologis dan Hasil Penelitian
15. Contoh Pemakaian (untuk suatu indikasi)

Format penulisan pada buku bisa dijadikan acuan sebagai pembangunan *database*. Berikut adalah data nama tumbuhan pada Atlas Tumbuhan Obat Indonesia Jilid 3:

Tabel 3.1. Atlas Tumbuhan Obat Jilid 3

NO	Nama Tumbuhan	Nama Lain
1	Alamanda	Allamanda cathartica L

2	Avokad	<i>Persea gratissima</i> Gaertn
3	Belimbing wuluh	<i>Averrhoa Bilimbi</i> L
4	Brotowali	<i>Tinospora crispa</i> L Miers
5	Bugenvil	<i>Bougainvillea glabra</i> Choicy
6	Buncis	<i>Phaseolus vulgaris</i> L.
7	Bunga Matahari	<i>Helianthus annuus</i> L
8	Daun Jinten	<i>Coleus Amboinicus</i> Lour
9	Ginkgo	<i>Ginkgo biloba</i> L.
10	Ginje	<i>Thevatia peruviana</i> [Pers.] K. Schum.)
11	Jali	<i>Coix lacryma-jobi</i> L. var. <i>ma-yuen</i> [Roman.] St
12	Jarak	<i>Ricinus communis</i> Linn.
13	Jarak Bali	<i>Jatropha Podagrica</i> Hook.
14	Jarak Pagar	<i>Jatropha curcas</i> L.
15	Jarong	<i>Achyranthes aspera</i> L.
16	Jure	<i>Nerium Indicum</i> Mill
17	Karet Kebo	<i>Ficus Elastica</i> Roxb
18	Kayu Putih	<i>Melaleuce Leucadendra</i> L
19	Kelapa	<i>Cocos Nucifera</i> L
20	Kemang Merak	<i>Caesalpinia pulcherrima</i> [L.] Sw.
21	Mengkudu	<i>Morinda citrifolia</i> L
22	Bayam	<i>Amaranthus tricolor</i> L
23	Sambiloto	<i>Andrographis paniculata</i> [Burm.f.] Nees
24	Beluntas	<i>Pluchea indica</i> [L.] Less.
25	Gandarusa	<i>Justicia gendarussa</i> Burm.f.
26	Brokoli	<i>Brassica oleracea</i> var. <i>italica</i>
27	Temulawak	<i>Curcuma xanthorrhiza</i> Roxb.
28	Wortel	<i>Daucus carota</i> L.
29	Meniran	<i>Phyllanthus niruri</i> L.
30	Salam	<i>Syzygium polyanthum</i> [Wight.] Walp
31	Kembang Telang	<i>Clitoria ternatea</i> L.

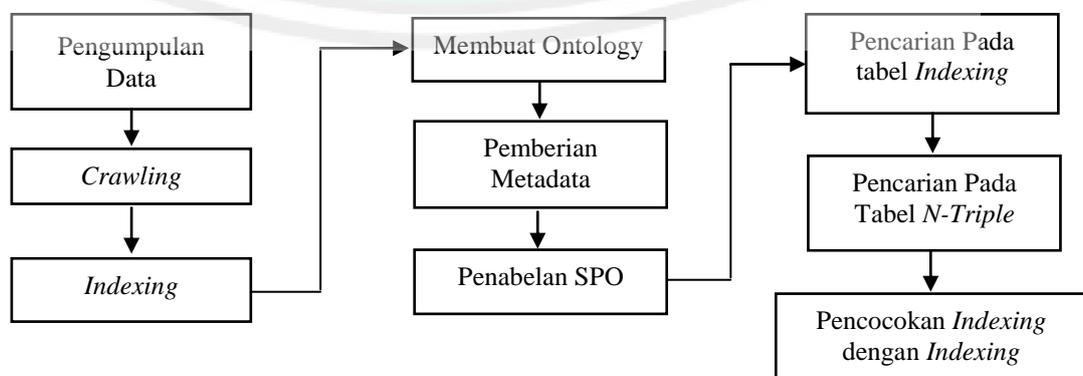
32	Ki Tolod	Isotoma longiflora PresL.
33	Kol Banda	Pisonia alba Span.
34	Legundi	Vitex trifolia L
35	Leunca	Solanum nigrum L.
36	Lidah Buaya	Aloe vera L
37	Memang Besar	Cleome spinosa L
38	Mentimun	Cucumis sativus L
39	Nangka	Artocarpus heterophyllus Lamk.
40	Pacing	Costus specios [Koenig] J.E. Smith,
41	Paku simpai	Cibotium barometz[L.] J.sm.
42	Pare	Momordica charantia L.
43	Patikan Kebo	Euphorbia hirta L
44	Pulutan	Urena lobata L.
45	Puring	Codiaeum variegatum [L.] BL
46	Rumput Mutiara	Hedyotis corymbosa[L.] Lamnk.
47	Saga	Abrus precatorius L
48	Selasih	Ocimum balisicium L
49	Semanggi Gunung	Hydrocotyle sibthorpioides Lam
50	Sereh	Cymbopogon nardus [L.] Rendle

3.2 Tahapan Proses *Semantic Web*

Pembangunan *semantic web* terbagi menjadi sembilan tahapan sebagai berikut :

- Pengumpulan data tentang tanaman obat yang ada di Indonesia Pada penelitian ini data diambil dari buku “Atlas Tumbuhan Indonesia Jilid 3”.

- *Crawling* yaitu proses mencari url yang terdapat pada domain dengan menggunakan aplikasi Sphider (Sphider, 2013). Domain yang digunakan pada penelitian ini yaitu <http://tobat.web-semantik.com>
- *Indexing* yaitu proses menghitung jumlah kata yang berhubungan dengan tanaman obat yang terdapat pada data.
- Membuat file ontology menggunakan aplikasi Protégé yang digunakan sebagai pemetaan data tentang tanaman obat.
- Membuat metadata file ontology yang telah dibangun menggunakan aplikasi Smoore.
- Penabelan SPO dan penyimpanan kedalam database menggunakan API JENA.
- Pencarian kata yang berhubungan dengan tanaman obat pada tabel *indexing*
- Pencarian kata yang berhubungan dengan tanaman obat pada tabel *n-triple*.
- Pencocokan Hasil Pencarian *Indexing* dengan pencarian *n-triple* menggunakan algoritma boyer-moore.



Gambar 3.3 Tahapan proses dalam membangun *semantic web*

3.3 Perancangan Aplikasi

Pada aplikasi ini yang diberi wewenang untuk mengolah data adalah administrator. *User* hanya dapat melihat tampilan data pada halaman web.

3.3.1 Desain aplikasi

Pada desain aplikasi ini memiliki dua macam hak akses yaitu :

1. Administrator

Admin memiliki wewenang data utama dan manajemen data tanaman, bagian tanaman, khasiat, bagian yang digunakan, cara pemakaian, dan penelitian.

2. User

User memiliki wewenang berupa pencarian dan informasi tanaman obat.

Relasi table untuk pembuatan aplikasi ensiklopedi berbasis mobile ini terdiri dari enam tabel yaitu :

- a. Tabel tanaman

Tabel_tanaman sebagai tempat penyimpanan data tentang identitas tanaman.

- b. Tabel bagian_tanaman

Tabel bagian_tanaman sebagai tempat penyimpanan nama bagian-bagian dari tanaman.

- c. Tabel khasiat

Tabel bagian_sebagai tempat penyimpanan suatu bagian dari suatu tanaman beserta kandungan kimianya.

d. Tabel cara_pemakaian

Tabel cara_pemakaian sebagai tempat penyimpanan cara pemakaian tanaman yang digunakan untuk mengobati suatu indikasi.

e. Tabel bagian_digunakan

Tabel bagian_digunakan sebagai tempat penyimpanan bagian tanaman yang digunakan.

f. Tabel penelitian

Tabel penelitian sebagai tempat penyimpanan penelitian tentang suatu tanaman.

Tabel 3.2 Tabel tanaman

NAME	CODE	TYPE	I	M
Idtanaman	IDTANAMAN	I	No	No
nama_indo	NAMA_INDO	VA45	No	No
nama_latin	NAMA_LATIN	VA45	No	No
Suku	SUKU	VA45	No	No
Sinonim	SINONIM	TXT	No	No
nama_daerah	NAMA_DAERAH	TXT	No	No
nama_asing	NAMA_ASING	TXT	No	No
nama_simplisia	NAMA_SIMPLISIA	TXT	No	No
deskripsi_tanaman	DESKRIPSI_TANAMAN	TXT	No	No
Catatan	CATATAN	TXT	No	No
Nama_url	NAMA_URL	VA40	No	No
Foto	FOTO	VA45	No	No

Tabel 3.3 Tabel bagian_tanaman

NAME	Code	Type	I	M
Idbagian	IDBAGIAN	I	No	No
nama_bagian	NAMA_BAGIAN	VA45	No	No

Tabel 3.4 Tabel bagian_digunakan

NAME	Code	Type	I	M
idbagian_digunakan	IDBAGIAN_DIGUNAKA N	I	No	No
Idtanaman	IDTANAMAN	I	No	No
Idbagian	IDBAGIAN	I	No	No
kandungan_kimia	KANDUNGAN_KIMIA	TXT	No	No

Tabel 3.5 Tabel khasiat

NAME	CODE	TYPE	I	M
Idkhasiat	IDKHASIAT	I	No	No
nama_penyakit	NAMA_PENYAKIT	VA45	No	No

Tabel 3.6 Tabel cara_pemakaian

NAME	CODE	TYPE	I	M
idcara_pemakaian	IDCARA_PEMAKAIAN	I	No	No
idbagian_digunakan	IDBAGIAN_DIGUNAKAN	I	No	No
Idkhasiat	IDKHASIAT	I	No	No
cara_pemakaian	CARA_PEMAKAIAN	TXT	No	No

Tabel 3.7 Tabel penelitian

NAME	CODE	TYPE	I	M
------	------	------	---	---

Idpenelitian	IDPENELITIAN	I	No	No
Peneliti	PENELITI	VA100	No	No
Tempat	TEMPAT	VA100	No	No
Tahun	TAHUN	I	No	No
Idtanaman	IDTANAMAN	I	No	No
Hasil	HASIL	TXT	No	No

Tabel 3.8 Tabel user

NAME	CODE	TYPE	I	M
Id	IDPENELITIAN	I	No	No
Username	PENELITI	VA40	No	No
Password	TEMPAT	VA40	No	No
Name	TAHUN	VA25	No	No

3.3.2 Desain Interface

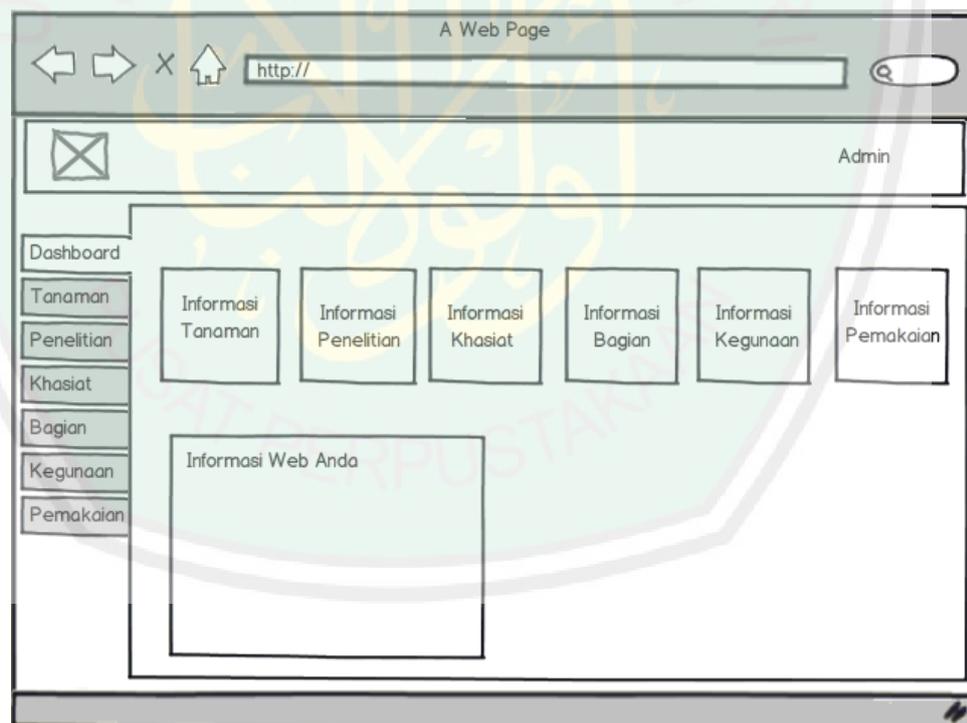
Pada aplikasi pencarian berbasis *mobile* ini memiliki 2 tampilan yang terdiri dari tampilan administrator dan tampilan user. Pada tampilan administrator ini adalah tampilan yang berfungsi untuk manajemen data tanaman obat. Bagian administrator ini berbasis *website*.

Tampilan user terdiri dari Halaman depan, halaman pencarian dan halaman informasi tanaman obat. Halaman depan berisi judul website, logo, Navigasi huruf berisi tombol A-Z dan navigasi tombol pencarian. Pada navigasi tombol A-Z, dimana setiap huruf yang ditekan akan menampilkan nama-nama tanaman berawalan huruf yang dipilih pada bagian informasi tanaman yang

dipilih. Setiap nama tanaman yang tampil jika ditekan akan menampilkan profil dari tanaman tersebut.

3.3.2.1 Halaman Awal Administrator

Pada tampilan untuk administrator terdiri dari halaman depan, halaman pengolahan data tanaman, halaman pengolahan data khasiat, halaman pengolahan data bagian tanaman, halaman pengolahan data bagian digunakan, halaman pengolahan data cara pemakaian dan halaman pengolahan data penelitian tanaman. Berikut ini adalah rancangan halaman depan dari administrator manajemen tanaman obat.



Gambar 3.4 Rancangan tampilan awal

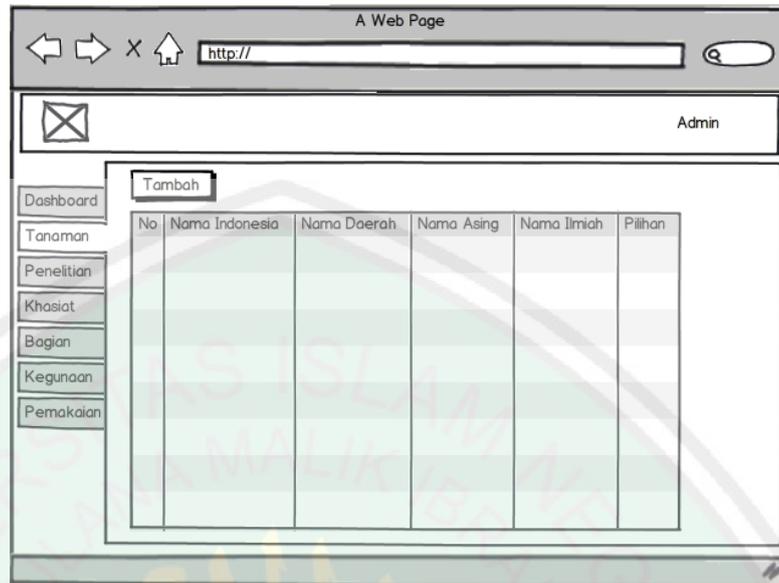
3.3.2.2 Halaman Data Tanaman

Pada halaman data tanaman administrator menginputkan data tanaman obat. Pada form input ini terdapat nama tanaman dengan bahasa Indonesia, nama suku, nama latin, sinonim, nama daerah, nama asing dan nama simplisia. Pada halaman data tanaman juga terdapat halaman view data tanaman obat yang telah disimpan. Berikut adalah rancangan halaman data tanaman.

The image shows a web browser window with the following elements:

- Browser title: A Web Page
- Address bar: http://
- User: Admin
- Navigation icons: Back, Forward, Home, Refresh
- Sidebar menu:
 - Dashboard
 - Tanaman
 - Penelitian
 - Khasiat
 - Bagian
 - Kegunaan
 - Pemakaian
- Main form title: FORM DATA TANAMAN
- Form fields:
 - Nama Indonesia :
 - Nama Latin :
 - Nama Suku :
 - Sinonim :
 - Nama Daerah :
 - Nama Asing :
 - Nama Simplisia :
- Buttons: Simpan, Reset

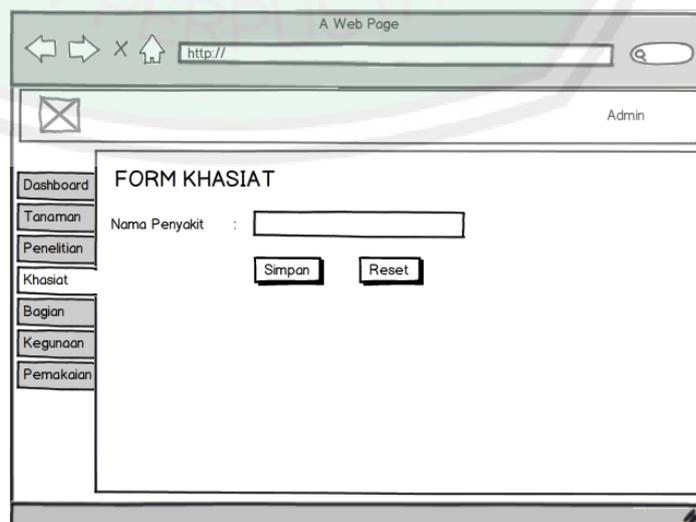
Gambar 3.5 Rancangan *input* data tanaman



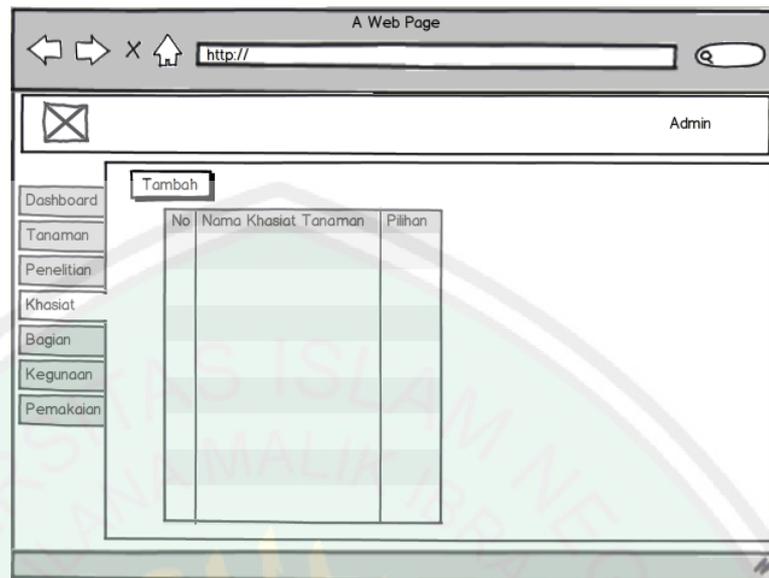
Gambar 3.6 Rancangan *view* data tanaman

3.3.2.3 Halaman Data Khasiat

Pada halaman data khasiat administrator menginput khasiat obat berdasarkan nama penyakit yang dapat disembuhkan dengan tanaman obat tersebut. Pada halaman data khasiat terdapat inputan nama penyakit dan view nama khasiat tanaman.



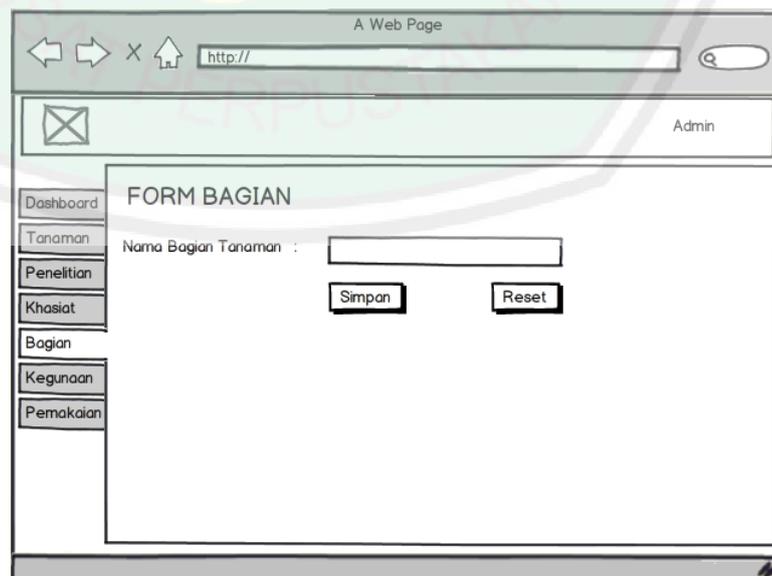
Gambar 3.7 Rancangan *input* data khasiat



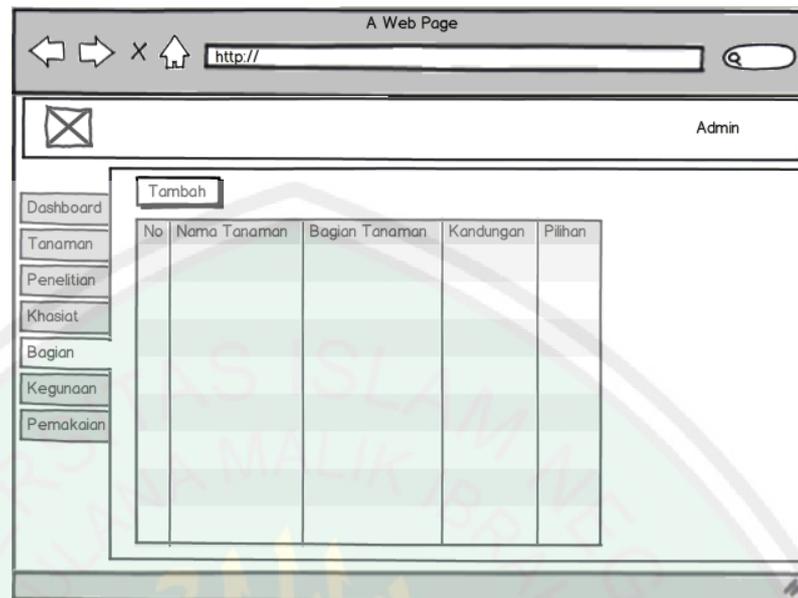
Gambar 3.8 Rancangan *view* data khasiat

3.3.2.4 Halaman Data Bagian Tanaman

Pada halaman data bagian tanaman administrator menginputkan naman bagian tanaman. Pada halaman data bagian tanaman terdapat input nama bagian dan view data nama bagian tanaman.



Gambar 3.9 Rancangan *input* bagian tanaman

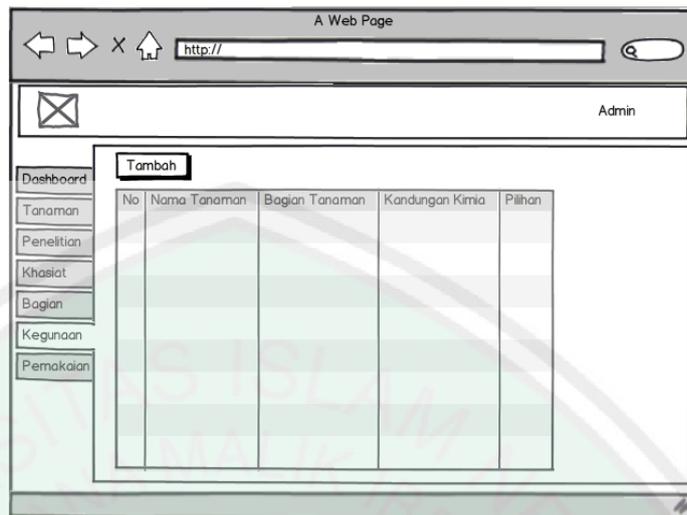


Gambar 3.10 Rancangan *view* data bagian tanaman

3.3.2.5 Halaman Data Bagian Digunakan

Pada halaman data bagian digunakan administrator menginputkan nama tanaman bagian tanaman dan kadungan kimia apa saja yang terdapat dalam tanaman tersebut. Pada halaman data bagian tanaman terdapat form input dan *view* data bagian tanaman yang digunakan.

Gambar 3.11 Rancangan *input* data bagian digunakan

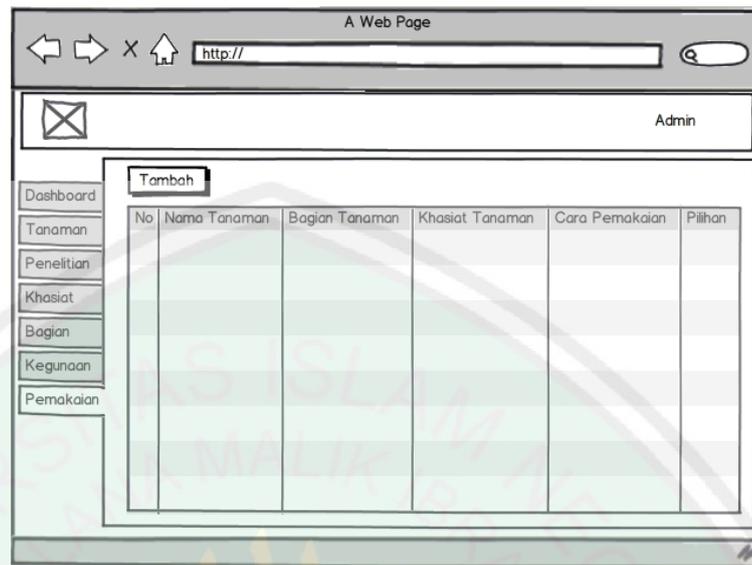


Gambar 3.12 Rancangan *view* data bagian digunakan

3.3.2.6 Halaman Data Cara Pemakaian

Pada halaman data cara pemakaian administrator menginputkan nama bagian tanaman yang digunakan, khasiat dari tanaman dan cara pemakaian tanaman obat. Pada halaman data cara pemakaian terdapat form input dan view data cara pemakaian.

Gambar 3.13 Rancangan *input* data pemakaian

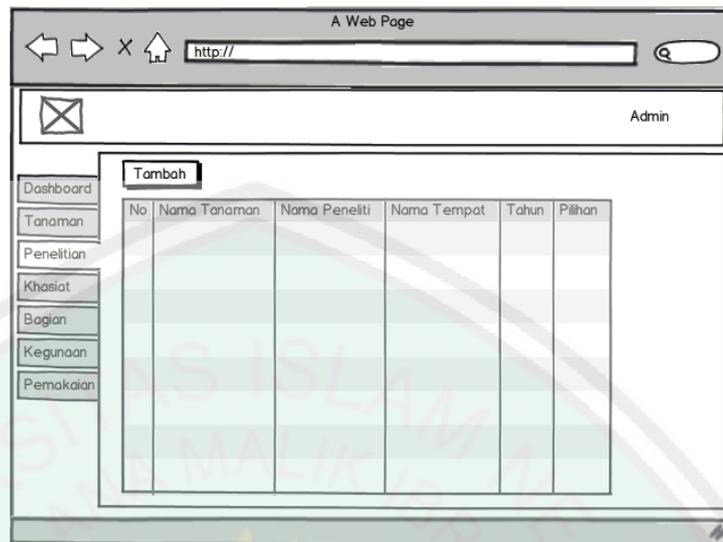


Gambar 3.14 Rancangan *view* data pemakaian

3.3.2.7 Halaman data penelitian

Pada halaman data penelitian administrator menginputkan nama peneliti, tempat penelitian, tahun peneliti, nama tanaman dan hasil penelitian dari tanaman obat. Pada halaman data penelitian terdapat form input dan view data penelitian.

Gambar 3.15 Rancangan *input* data penelitian



Gambar 3.16 Rancangan *view* data penelitian

3.3.2.8 *Interface Awal Aplikasi Mobile*

Pada tampilan user terdiri dari halaman depan, halaman pencarian, dan Halaman Data Tanaman A-Z. Berikut ini adalah rancangan halaman-halamannya.



Gambar 3.17 Rancangan tampilan awal

3.3.2.9 Interface Pencarian

Pada halaman halaman pencarian ini adalah langsung menetik nama tanaman obat apa yang ingin dicari. Pada halaman ini user menetik kata-kata yang berkaitan dengan tanaman obat kemudian menekan tombol cari. Setelah menekan tombol cari maka akan muncul informasi tanaman obat yang dicari.



Gambar 3.18 Rancangan tampilan cari tanaman



Gambar 3.19 Rancangan tampilan Hasil cari

3.3.2.10 *Interface Data Tanaman A-Z*

Halaman data tanaman A-Z ini terdapat list tanaman dengan huruf depan nama tanaman A-Z. pada halaman ini akan memberikan informasi nama tanaman beserta manfaatnya dari A-Z. Jika ingin mengetahui nama tanaman obat yang berhuruf A maka menekan list A dan tanaman yang memiliki nama depan A akan muncul. Kemudian ingin mengetahui informasi nama tanaman tersebut maka user akan menekan nama tanaman. Setelah itu maka akan muncul informasi nama tanaman sesuai yang dipilih user.



Gambar 3.20 Rancangan tampilan menu data tanaman A-Z



Gambar 3.21 Rancangan tampilan data tanaman A-Z

Gambar 3.22 Rancangan tampilan informasi tanaman

3.4 Perancangan Ontologi

Beberapa tahapan perancangan *ontology* adalah sebagai berikut :

- Menentukan ruang lingkup *ontology*
- Mendefinisikan kelas dan tingkatan kelas
- Definisi Properti class dan faset

- Membuat Instance

3.4.1 Rancangan Domain Dan Ruang Lingkup Ontologi

Perancangan domain dan ruang lingkup ontologi yaitu mempresentasikan seluruh dokumen yang dilengkapi dengan informasi dan dikelompokkan dalam sebuah *class*. Domain dalam pembuatan aplikasi ini adalah tanaman dengan ruang lingkup ontologi sebagai berikut:

- *Class* tanaman,
- *Class* penelitian,
- *Class* pemakaian,
- *Class* nama (memiliki *subclass* asing, *subclass* daerah, *subclass* latin, *subclass* simplisia, *subclass* sinonim dan *subclass* suku),
- *Class* khasiat,
- *Class* kandungan_kimia,
- *Class* bagian_tanaman (memiliki *subclass* air, *subclass* akar, *subclass* batang, *subclass* biji, *subclass* buah, *subclass* bunga, *subclass* daun, *subclass* getah, *subclass* herba, *subclass* kayu, *subclass* minyak, *subclass* rimpang, *subclass* tangkai dan *subclass* tempurung).

3.4.2 Mendefinisikan Kelas Dan Tingkat Kelas

Pada tahap ini dilakukan pengelompokan kata ke dalam tingkatan kelas berdasarkan kesamaan karakteristik. Terdapat tiga cara dalam pembentukan tingkatan kelas, yakni: melalui proses pengembangan *top-down*, *botton-up*, atau melalui kombinasi antar kedua proses. Pada penelitian ini, proses pengembangan

dilakukan dengan cara pengembangan *top-down* dimana pendefinisian tingkatan kelas dimulai dari yang paling umum sampai ke yang lebih khusus. Dan tiap-tiap *class* yang akan dibuat akan menjadi *subclass* dari *class* “:THING”.

Perancangan *class* dan *subclass* tersebut dapat dilakukan dengan menggunakan perangkat lunak Protégé 3.4.5 seperti yang terlihat pada gambar 3.23.



Gambar 3.23: Rancangan Pendefinisian *Class* dan Tingkatan *Class*

3.4.3 Mendefinisikan Properti Kelas Dan Faset Dari Slot

Properti atau yang biasa disebut dengan *slot* merupakan hubungan antar objek. Properti bisa berupa kata kerja atau kata sifat. Sedangkan faset merupakan informasi tentang properti berupa nilai yang dimiliki oleh properti atau tipe nilai sebuah properti, seperti *string*, *number*, *boolean*, dan lain-lain.

Tabel 3.9: Rancangan properti pada *class* tanaman

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
mempunyai_bagian	Multiple bagian_tanaman	Instances	Object Property
mempunyai_nama_asing	Multiple asing	Instances	Object Property
mempunyai_nama_daerah	Multiple daerah	Instances	Object Property
mempunyai_nama_latin	Multiple latin	Instances	Object Property
mempunyai_nama_simplisia	Multiple simplisia	Instances	Object Property
mempunyai_sinonim	Multiple sinonim	Instances	Object Property
mempunyai_suku	Multiple suku	Instances	Object Property

Tabel 3.10: Rancangan properti pada *class* penelitian

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
penelitian_dari	Multiple tanaman	Instances	Object Property
Peneliti	Single String	Instances	Datatype Property
hasil_penelitian	Single String	Instances	Datatype Property

Tabel 3.11: Rancangan properti pada *class* pemakaian

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
penggunaan_dari	Multiple bagian_tanaman	Instances	Object Property
cara_pemakaian	Single String	Instances	Datatype Property

Tabel 3.12: Rancangan properti pada *subclass* asing *class* nama

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
nama_asing_dari	Multiple tanaman	Instances	Object Property
nama_asing	Single String	Instances	Datatype Property

Tabel 3.13: Rancangan properti pada *subclass* daerah *class* nama

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
nama_daerah_dari	Multiple tanaman	Instances	Object Property
nama_daerah	Single String	Instances	Datatype Property

Tabel 3.14: Rancangan properti pada *subclass* latin *class* nama

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
nama_latin_dari	<i>Multiple</i> tanaman	<i>Instances</i>	<i>Object Property</i>
nama_latin	<i>Single String</i>	<i>Instances</i>	<i>Datatype Property</i>

Tabel 3.15: Rancangan properti pada *subclass* simplisia *class* nama

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
nama_simplisia_dari	<i>Multiple</i> tanaman	<i>Instances</i>	<i>Object Property</i>
nama_simplisia	<i>Single String</i>	<i>Instances</i>	<i>Datatype Property</i>

Tabel 3.16: Rancangan properti pada *subclass* sinonim *class* nama

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
nama_sinonim_dari	<i>Multiple</i> tanaman	<i>Instances</i>	<i>Object Property</i>
nama_sinonim	<i>Single String</i>	<i>Instances</i>	<i>Datatype Property</i>

Tabel 3.17: Rancangan properti pada *subclass* suku *class* nama

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
nama_suku_dari	<i>Multiple</i> tanaman	<i>Instances</i>	<i>Object Property</i>
nama_suku	<i>Single String</i>	<i>Instances</i>	<i>Datatype Property</i>

Tabel 3.18: Rancangan properti *class* khasiat

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
penggunaan_dari	<i>Multiple</i> bagian_tanaman	<i>Instances</i>	<i>Object Property</i>
nama_penyakit	<i>Single String</i>	<i>Instances</i>	<i>Datatype Property</i>

Tabel 3.19: Rancangan properti *class* kandungan_kimia

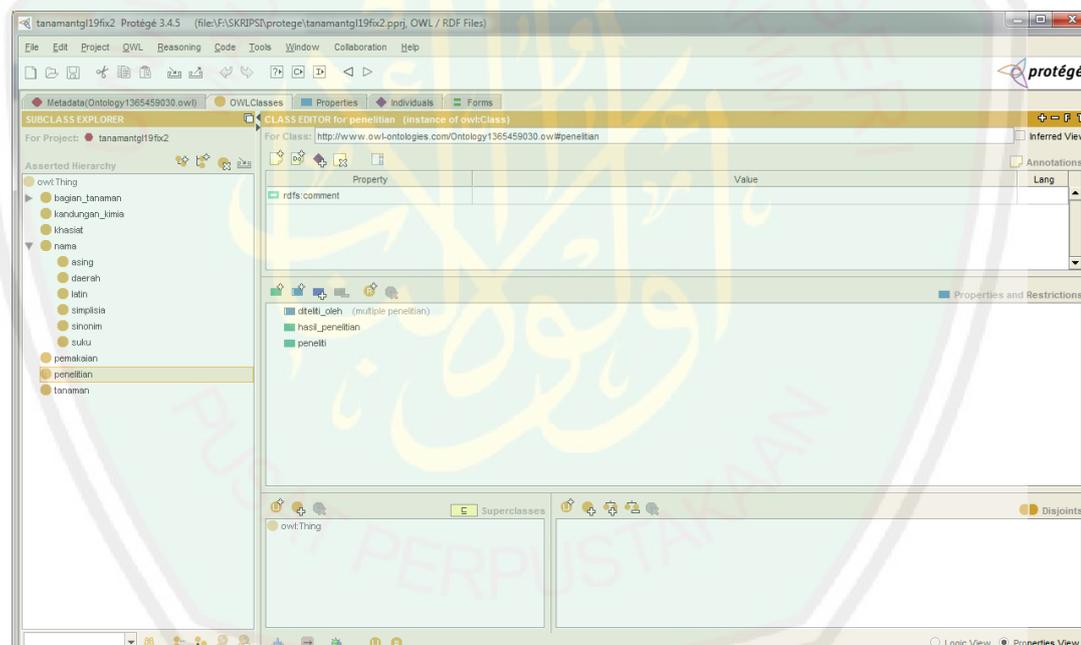
<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
kandungan_kimia_dari	<i>Multiple</i> bagian_tanaman	<i>Instances</i>	<i>Object Property</i>
nama_kandungan_kimia_	<i>Single String</i>	<i>Instances</i>	<i>Datatype Property</i>

Tabel 3.20: Rancangan properti pada *class* bagian_tanaman

<i>Property</i>	<i>Range</i>	<i>Allowed Value</i>	<i>Type</i>
bagian_dari	<i>Multiple</i> tanaman	<i>Instances</i>	<i>Object Property</i>
pemakaian_dari	<i>Multiple</i> pemakaian	<i>Instance</i>	<i>Object Property</i>

digunakan_untuk	<i>Multiple</i> khasiat	<i>Instance</i>	<i>Object Property</i>
kandungan_kimia_dari	<i>Multiple</i> kandungan_kimia	<i>Instance</i>	<i>Object Property</i>

Dari rancangan properti dan *faset* dari slot diatas dapat dibuat dengan menggunakan perangkat lunak Protégé 3.4.5. Pada tiap-tiap *subclass* yang telah dibuat diatas ditambahkan *object* dan *datatype property* pada “Properties View” dalam tab menu “OWL Classes”. Contoh hasil pembuatan properti pada *class* Penelitian dapat dilihat pada gambar 3.24.

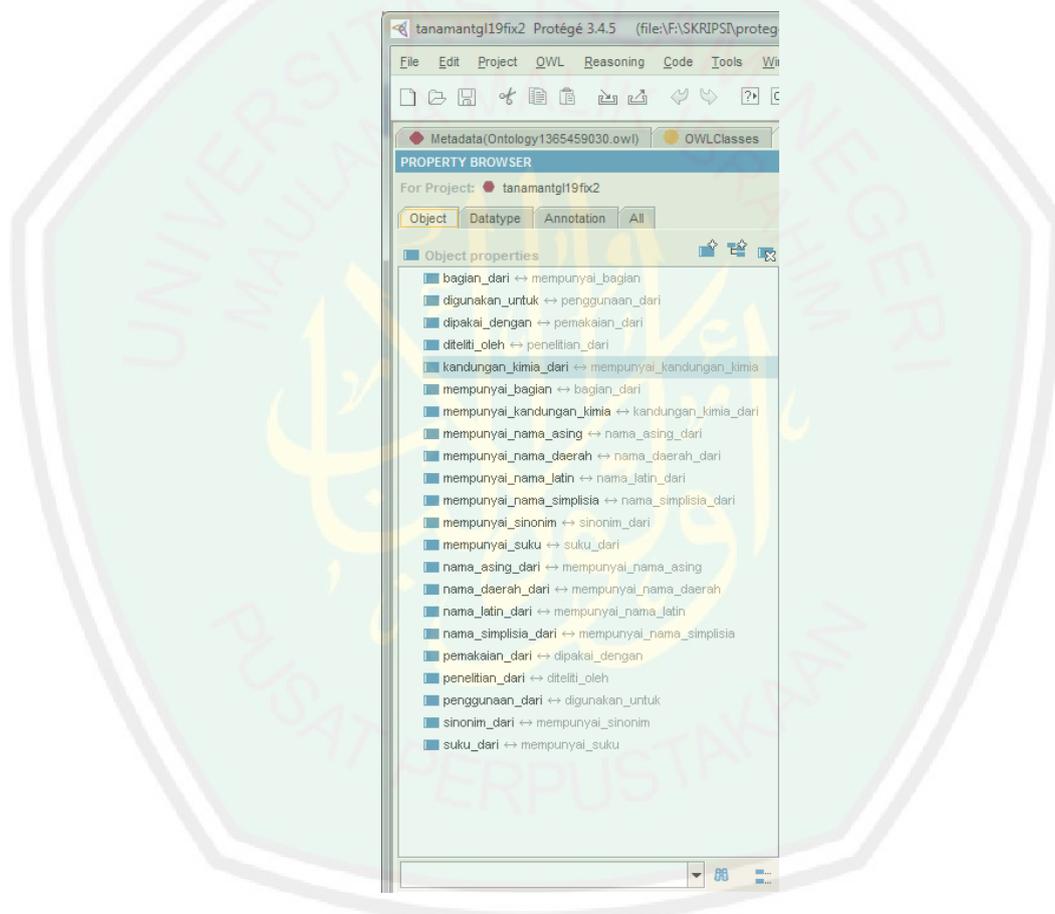


Gambar 3.24: Properti-Properti yang terbentuk pada *Subclass* penelitian

Selanjutnya masing-masing *object* dan *datatype property* yang telah dibuat secara keseluruhan akan otomatis terisikan pada “Object” dan “Datatype” pada tab menu “Properties” yang selanjutnya dilakukan pendefinisian fungsi *Inverse property* pada masing-masing *object property* yang telah terbentuk. *Inverse*

property ini bertujuan untuk memberikan nilai dari suatu properti secara otomatis jika *inverse property* diberi suatu nilai.

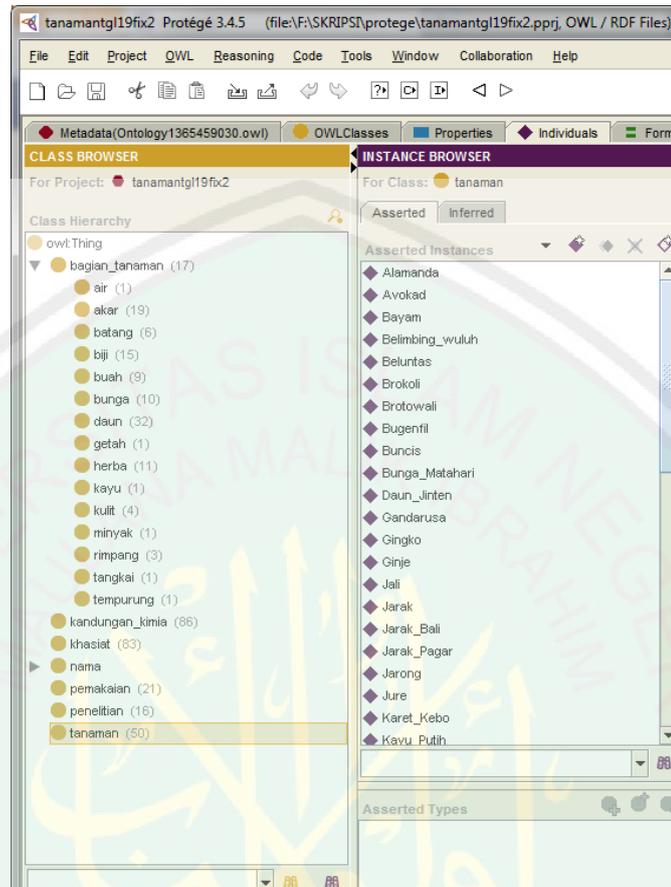
Sebagai contoh pada rancangan properti diatas, terdapat fungsi *inverse* pada *class* penelitian yakni pada properti “penelitian_dari” yang memiliki *inverse* “diteliti_oleh” pada *class* tanaman(gambar 3.25).



Gambar 3.25: Pendefinisian Fungsi *Inverse* pada Tiap *Object Property*

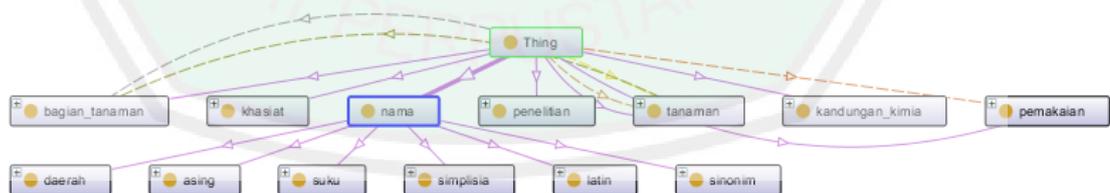
3.4.4 Membuat *Instance*

Instance merupakan objek maupun individual dari tiap-tiap kelas yang telah terbentuk sebelumnya. Pembentukan *instance* dapat dibuat melalui tab menu “Individuals” dalam Protégé. Contoh hasil pembuatan *instance* dari kelas pendiri dengan *subclass* tanaman dapat dilihat pada gambar 3.26.



Gambar 3.26: Individual-Individual yang Terdapat pada tanaman

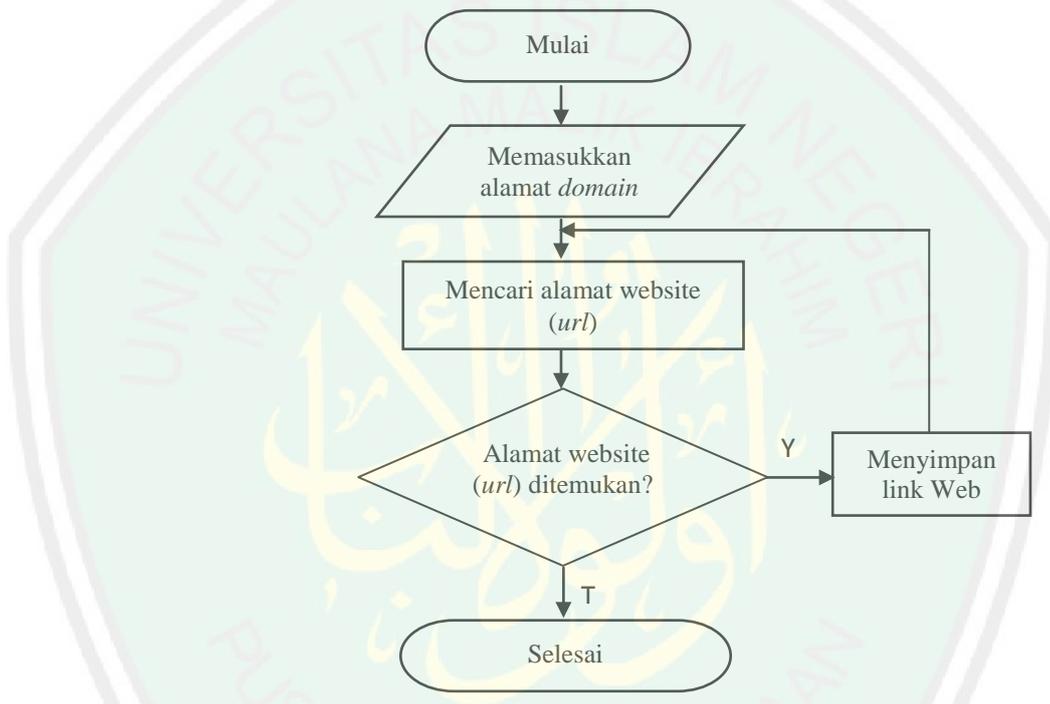
3.4.5 Skema Graph Ontology



Gambar 3.27: Skema *Ontology* Tanaman Obat

3.5 Crawling Menggunakan Sphider

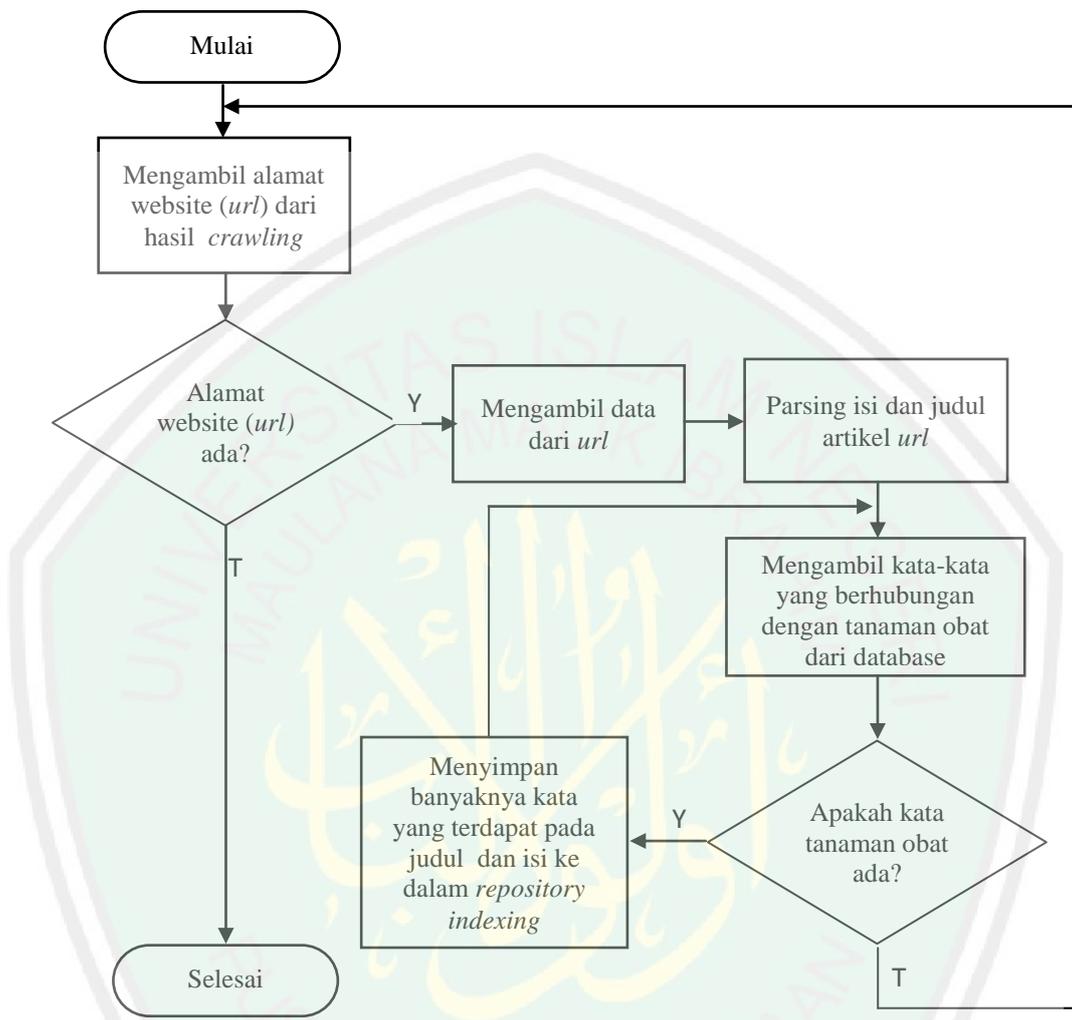
Pada proses *crawling* ini akan dilakukan penelusuran halaman-halaman website yang terdapat pada *domain* website untuk mencari link terkait, hasil *crawling* yang berupa link akan disimpan ke dalam database.



Gambar 3.28: Proses Alur Crawling pada sphider

3.6 Indexing Kata Dari Halaman Website

Proses lanjutan setelah *crawling* adalah *indexing*. Pada Penelitian ini adalah *indexing* akan dilakukan untuk mencari kata-kata yang berhubungan dengan tanaman obat pada halaman-halaman website hasil *crawling*. Pada setiap halaman akan dilakukan proses *parsing* untuk membedakan antara judul dan isi. Hasil *indexing* yaitu jumlah kata yang berhubungan dengan tanaman obat yang berada pada judul dan isi yang akan disimpan ke dalam *database*.

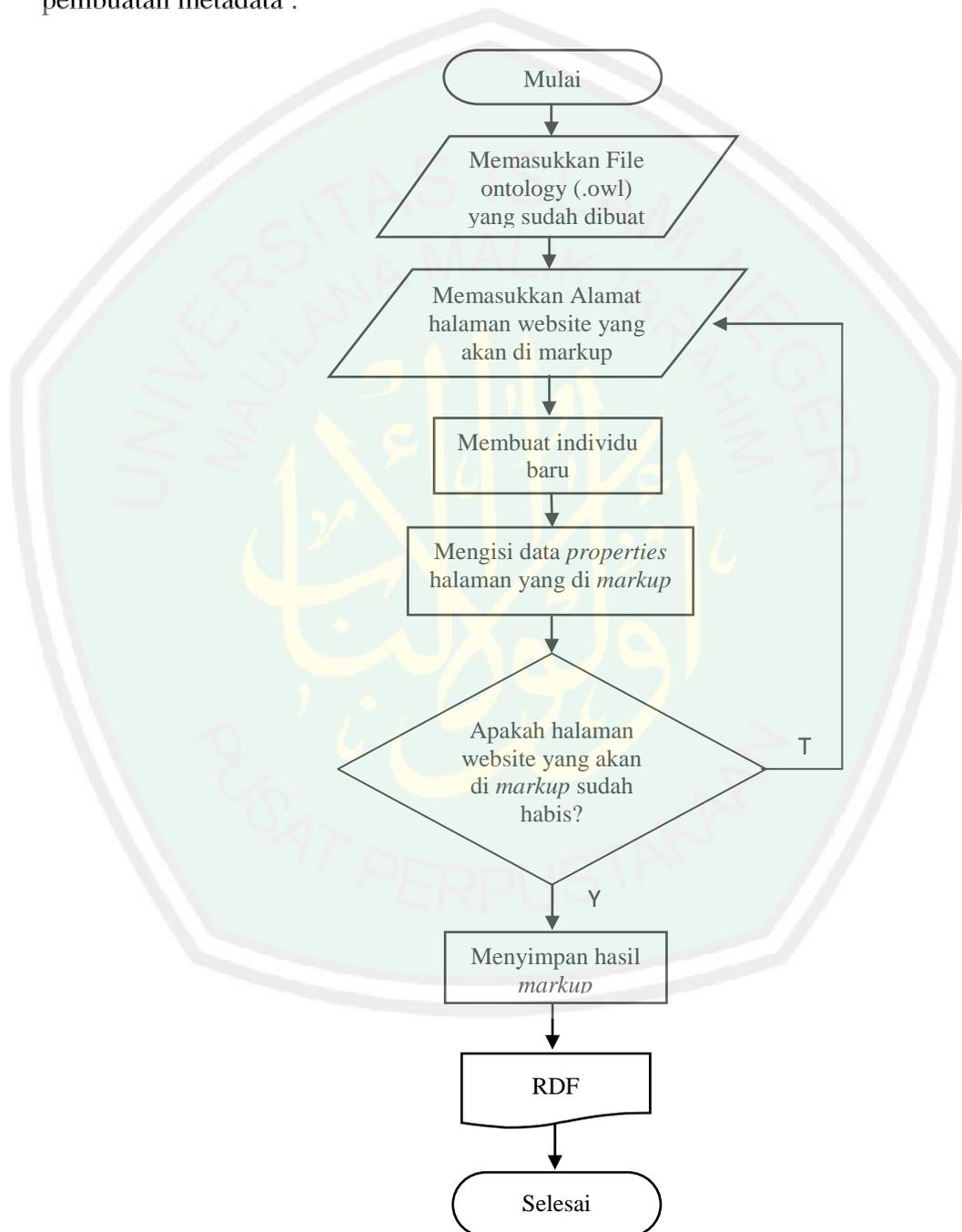


Gambar 3.29: Proses Alur Indexing Halaman web

3.7 Markup Halaman Website Menggunakan Smore

Markup halaman website adalah proses penambahan metadata pada setiap halaman website, data hasil *markup* akan dipergunakan untuk pembuatan *N-triple table* yang terdiri dari subjek, predikat, dan objek. Pada penelitian ini untuk melakukan *markup* akan menggunakan perangkat lunak Smore. Pada masing-masing halaman website dibuat individual baru pada menu “New Individuals”, penamaan individual menggunakan format nama tanaman dan diikuti dengan

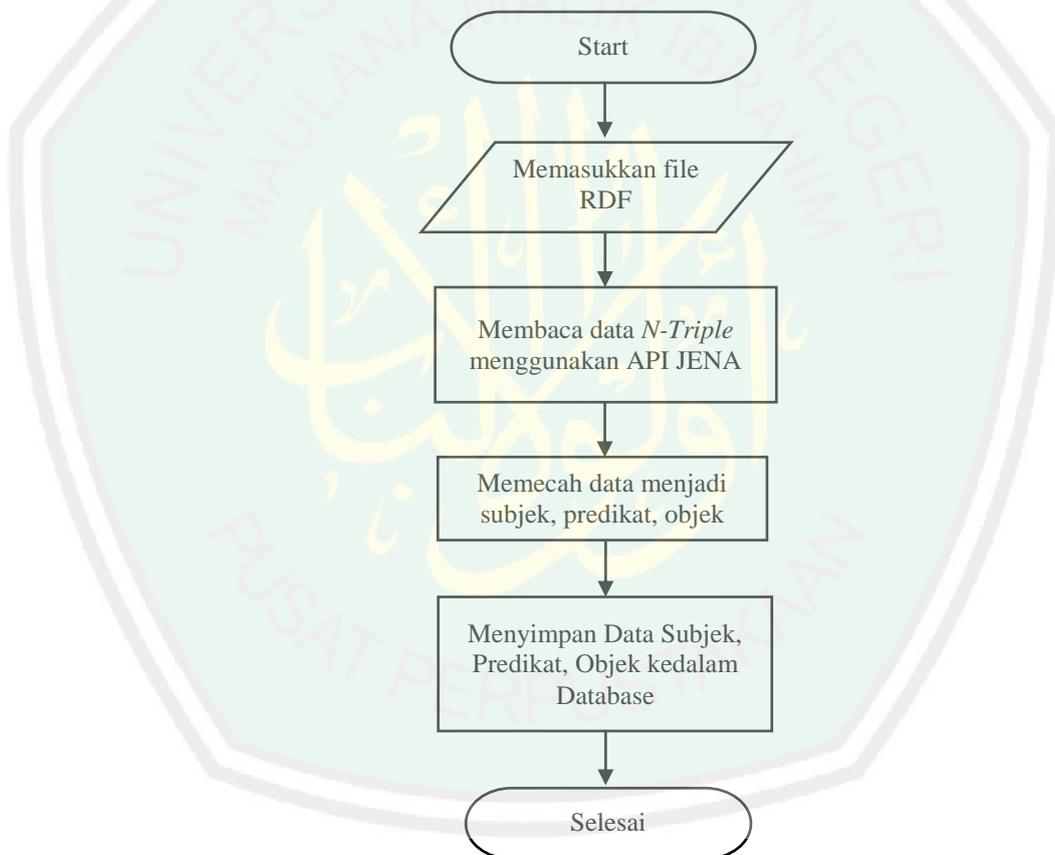
nama bagian tanaman, Setiap individual diisikan objek propertis dan data propertis sesuai dengan rancang ontologi yang telah dibuat. Berikut alur pembuatan metadata :



Gambar 3.30: Alur Proses Smore

3.8 Menabelkan *N-Triple* Menggunakan API JENA

Pada tahap selanjutnya adalah Penabelan *N-Triple* menggunakan API JENA. API JENA disini digunakan untuk memarsing data hasil *smoore*. Data RDF hasil *smoore* akan dipisahkan antara subjek, predikat dan objeknya untuk disimpan dalam *database*.

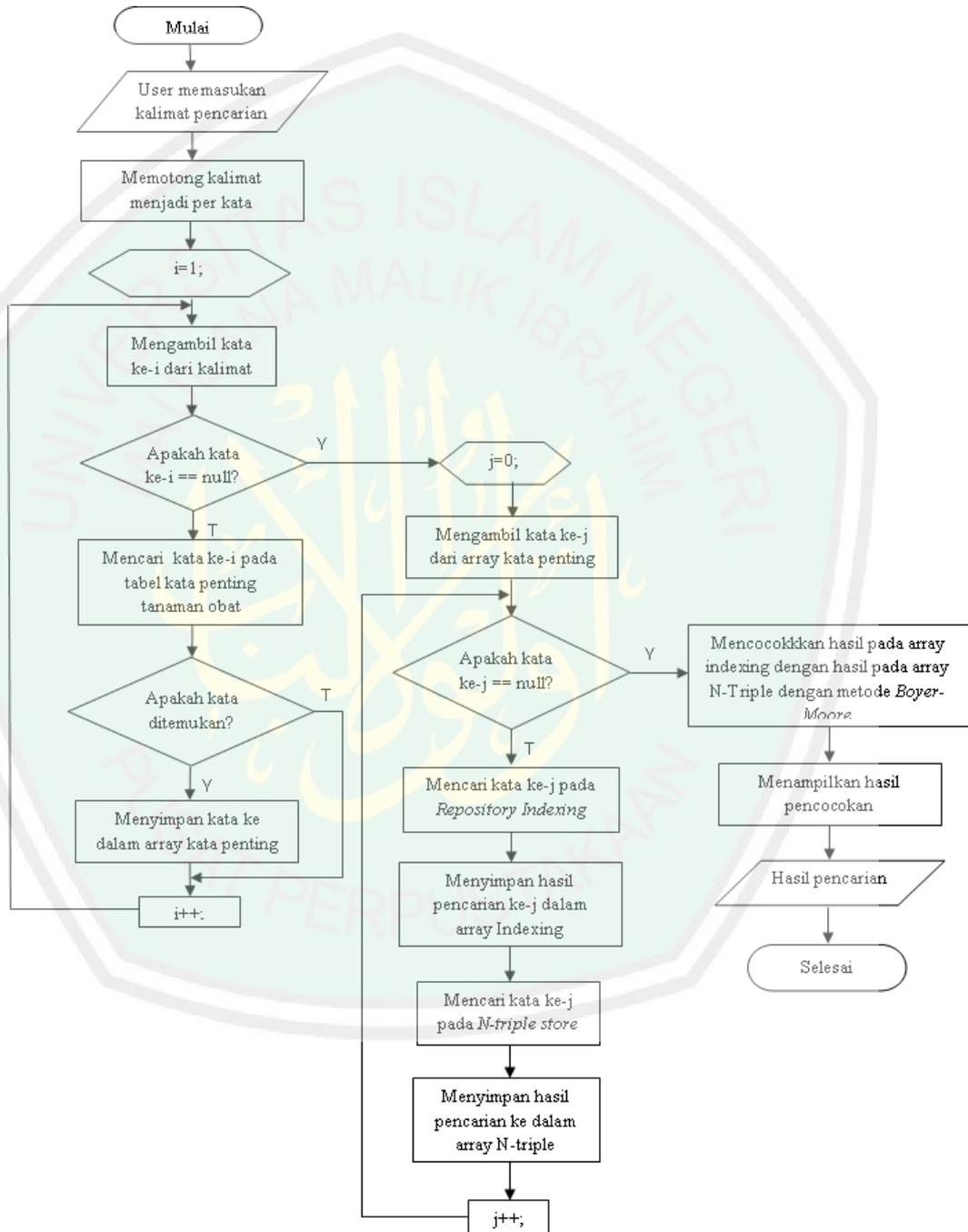


Gambar 3.31: Alur Proses Penabelan N-triple

3.9 Proses Pencarian

Pada proses pencarian, kalimat yang di masukkan oleh user akan di parsing dan dicari kata-kata yang berhubungan dengan tanaman obat, selanjutnya kata yang berhubungan dengan tanaman obat akan di cari pada *Indexing Store* dan

N-Triple Store. Hasil dari pencarian dari *Indexing Store* dan *N-Triple Store* akan dicocokkan menggunakan metode *boyer moore*.

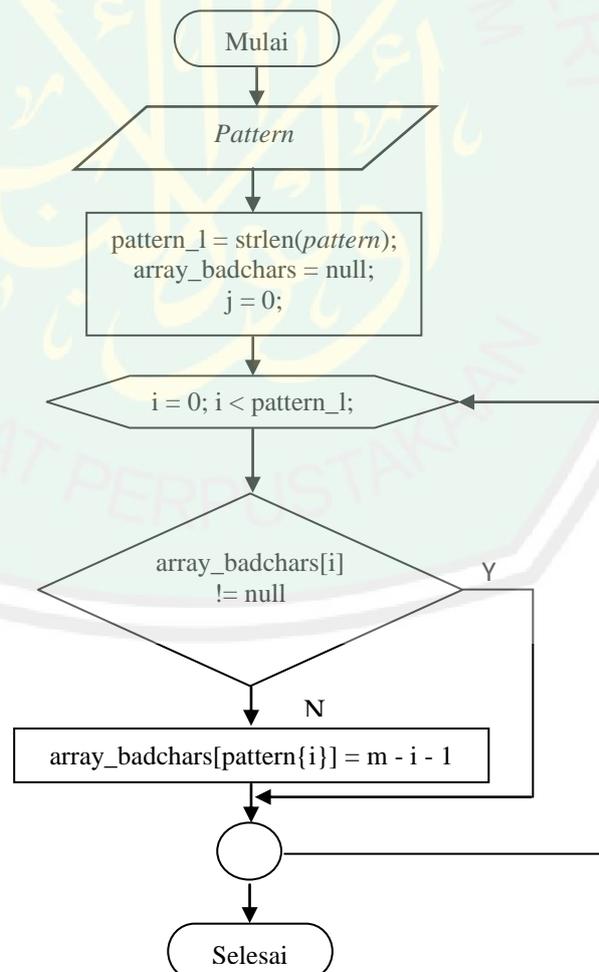


Gambar 3.32: Alur Proses pencarian

3.10 Proses Pencocokan Dengan Metode Boyer-Moore

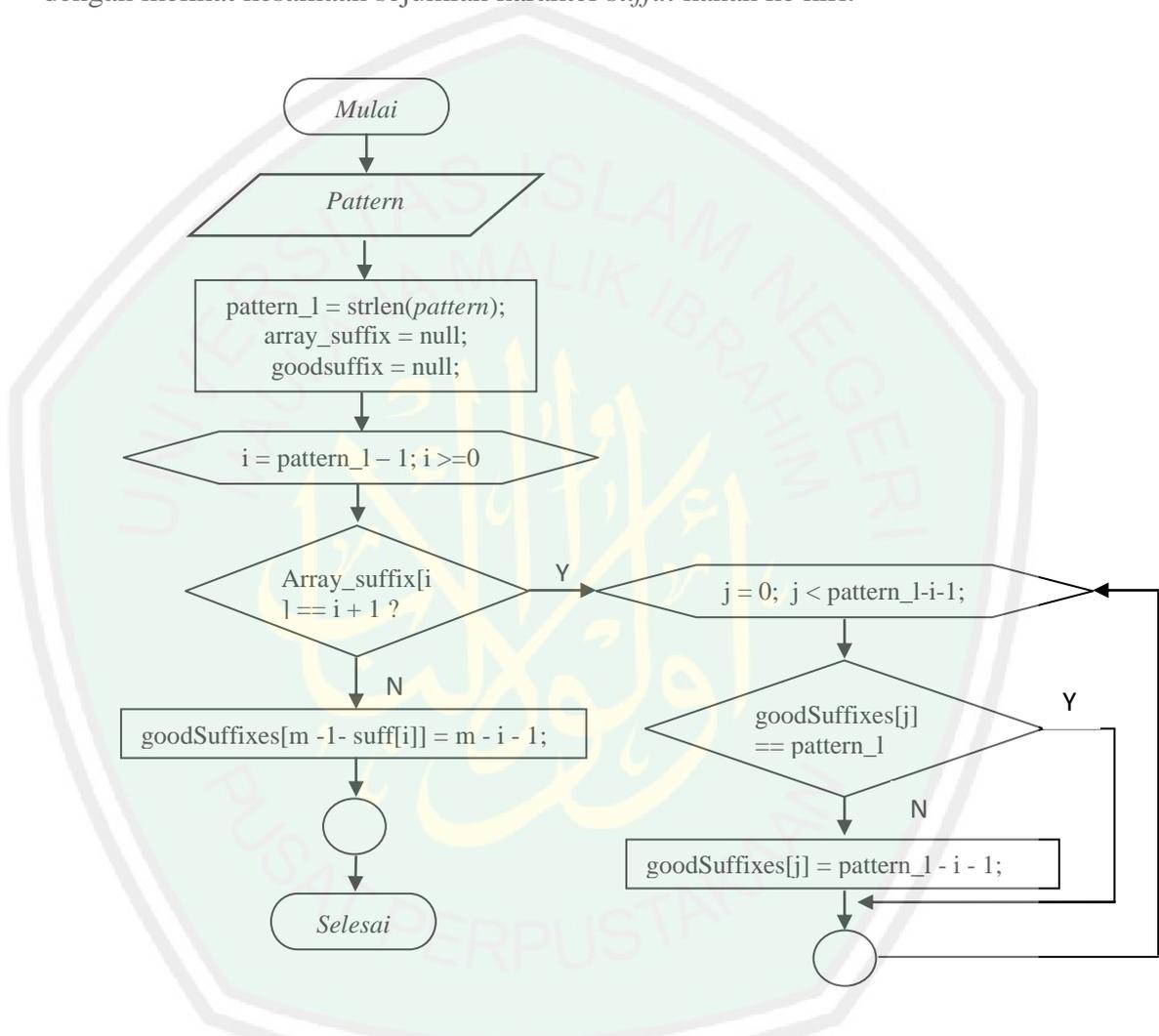
Hasil pencarian dari *indexing store* dicocokkan hasil pencarian dari *ntriple store* dengan menggunakan boyer-moore untuk perangkian. Pada Algoritma Boyer-Moore sebelum proses pencarian dilakukan harus dibentuk tabel *Occurence Heuristic (Bad Character)* dan *Match Heuristic (Good Suffix)* terlebih dahulu yang nantinya akan digunakan sebagai nilai pergeseran.

Pemberian nilai Tabel *Occured Heuristic* dimulai dari *pattern* paling kanan, *pattern* paling akhir diberi nilai 0 selanjutnya dilakukan pergeseran ke kiri untuk pencocokan karakter *pattern*.

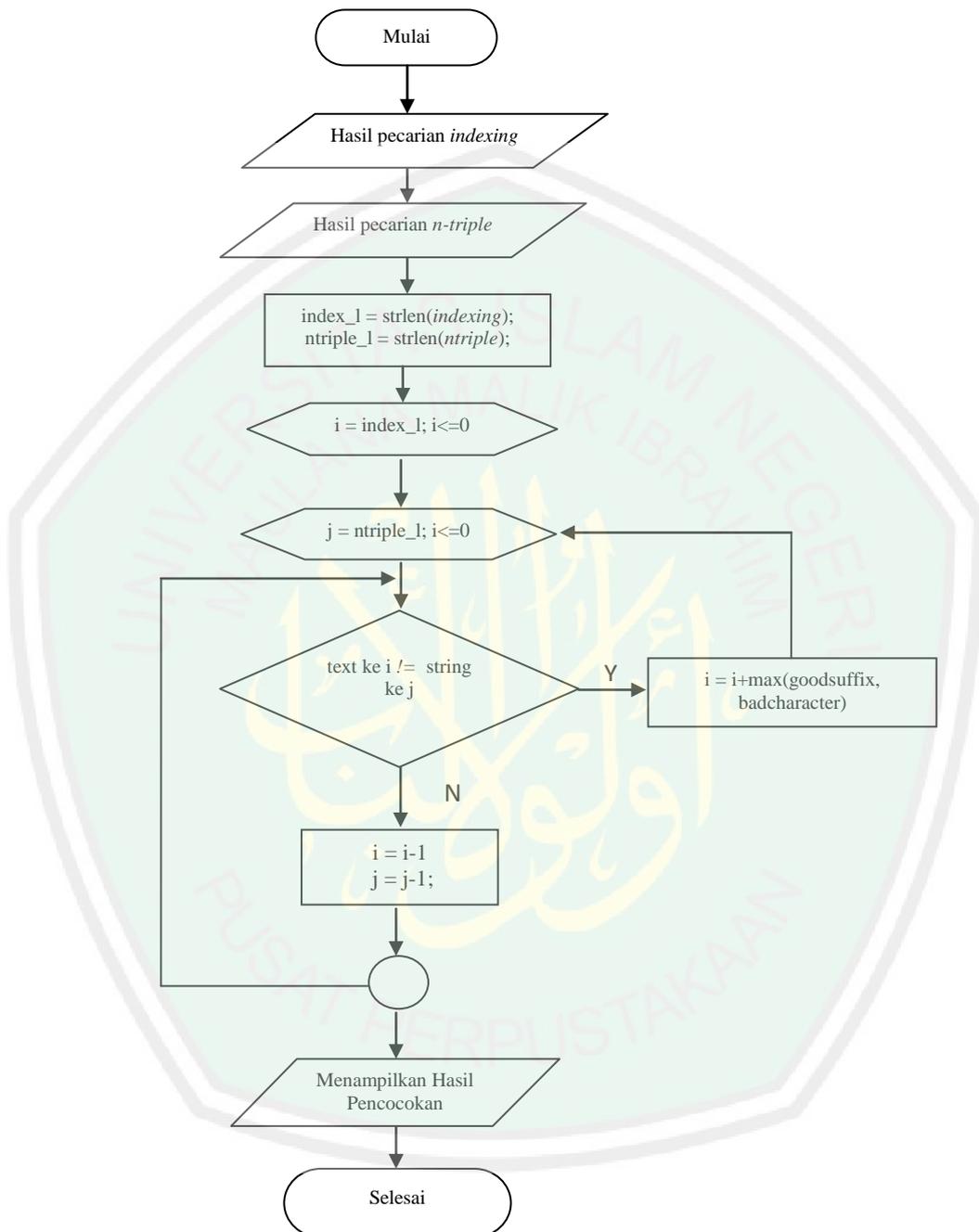


Gambar 3.33: Alur Pembentukan Tabel *Occurence Heuristic*

Pembentukan suffix dari *pattern* dimulai kanan ke kiri. Pergeseran *Suffix* diberi nilai 1. Pemberian nilai MH pada karakter-karakter sebelumnya dilakukan dengan melihat kesamaan sejumlah karakter *suffix* kanan ke kiri.



Gambar 3.34: Alur Pembentukan Tabel *Match Heuristic*



Gambar 3.35: Alur Pencocokan dengan Algoritma Boyer Moore

Simulasi perhitungan manual pencocokan kata pada penelitian ini adalah sebagai berikut :

Hasil pencarian n-triple : alamanda

Hasil pencarian indexing : <http://tobat.web-semantic.com/alamanda.html>

Tabel 3.21: Pembentukan *suffix*

Suffix	Prefix	Nilai Pergeseran
null	A	8
a	L	7
al	A	6
ala	M	5
alam	A	4
alama	N	3
alaman	D	2
alamand	A	1

Tabel 3.22: Tabel pergeseran OH dan MH

Posisi	1	2	3	4	5	6	7	8
String	a	l	A	m	a	N	d	a
OH	0	6	0	4	0	2	1	0
MH	7	7	7	7	7	7	7	1

Tabel 3.23: Tabel *iterasi* pertama

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43				
h	t	t	p	:	/	/	t	o	b	a	t	.	w	e	b	-	s	e	m	a	n	t	i	k	.	c	o	m	/	a	l	a	m	a	n	d	a	.	h	t	m	l				
a	l	a	m	a	n	d	a																																							

- “t” tidak cocok dengan “a”.
- tabel OH : karakter “t” tidak ditemukan pada tabel OH sehingga pergeseran menggunakan nilai maksimal = 8.
- tabel MH : karakter “a” nilai pergeserannya = 1.
- sehingga geser string sebesar 8 posisi (nilai maksimal dari kedua pergeseran).

Tabel 3.24: Tabel *iterasi* kedua

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43					
h	t	t	p	:	/	/	t	o	b	a	t	.	w	e	b	-	s	e	m	a	n	t	i	k	.	c	o	m	/	a	l	a	m	a	n	d	a	.	h	t	m	l					
							a	l	a	m	a	n	d	a																																	

- “b” tidak cocok dengan “a”.
- tabel OH : karakter “b” tidak ditemukan pada tabel OH sehingga pergeseran menggunakan nilai maksimal = 8.
- tabel MH : karakter “a” nilai pergeserannya = 1.
- sehingga geser string sebesar 8 posisi (nilai maksimal dari kedua pergeseran).

3.11 Rancangan Uji Coba Aplikasi

Uji coba dilakukan untuk mengetahui apakah hasil pencarian dari aplikasi yang dibuat akurat dan cepat.



Gambar 3.36 Rancangan uji coba aplikasi

Aplikasi Pencarian di *Install* pada *smartphone* dengan OS Android 2.3.6 versi Gingerbread.dxlb1. Setelah itu melakukan pencarian tanaman obat dengan mengakses database pada server. Penghubung aplikasi pencarian tanaman obat dengan data *database* yaitu menggunakan *JSON*.

3.12 Kebutuhan sistem

Kebutuhan sistem yang dibutuhkan untuk membangun aplikasi mesin pencarian ini terbagi menjadi 2 yaitu sebagai berikut :

3.12.1 Perangkat Untuk Pembuatan Aplikasi

1. Hardware

Perangkat keras yang mendukung untuk proses pembangunan aplikasi ini adalah laptop atau komputer yang memiliki spesifikasi sebagai berikut :

- Processor 2.10GHz
- RAM 2.00 GB(1.87 GB usable)
- Hardisk 250 GB

2. Software

Perangkat lunak yang mendukung untuk proses pembangunan aplikasi ini yaitu :

- Windows 7 Ultimate (32-bit)
- Java JDK 1.6.0
- NetBeans IDE 7.0.1
- Eclipse Juno 4.2.1
- XAMPP 1.7.2
- Protégé 3.4.5
- Smore 5.0
- Notepad++
- Mozilla Firefox Browser

3.12.2 Perangkat Untuk Menjalankan Aplikasi

1. Hardware

Perangkat Keras yang mendukung untuk menjalankan aplikasi ini adalah *smartphone* yang memiliki spesifikasi sebagai berikut :

- Samsung Galaxy
- 830 MHz ARMv6
- 180 MB, 290 MB RAM

2. Software

Perangkat lunak yang mendukung untuk menjalankan aplikasi yaitu :

- OS Android 2.3.6 versi Gingerbread.dxlbl1.

BAB IV

HASIL DAN PEMBAHASAN

Dalam bab ini dibahas mengenai hasil uji coba sistem yang telah dirancang dan dibuat. Uji coba dilakukan untuk mengetahui apakah sistem dapat berjalan sebagaimana mestinya dengan lingkungan uji coba yang telah ditentukan sesuai dengan skenario yang dijelaskan pada bab 3. Uji coba pada aplikasi ini dilakukan untuk mengetahui seberapa cepat dan akurat dalam pencarian.

4.1 Aplikasi Halaman Admin Menggunakan PHP

Pada halaman admin data akan diakses secara online pada domain "<http://tobat.semantik-web.com/>". Pada halaman administrator terdapat beberapa halaman menu yaitu halaman depan administrator, halaman data tanaman, halaman input data tanaman, halaman detail tanaman, halaman data bagian tanaman, halaman input bagian tanaman, halaman data khasiat, halaman input khasiat, halaman data kandungan kimia, halaman input kandungan kimia, halaman data cara pemakaian, halaman input cara pemakaian, halaman data penelitian, halaman input penelitian, halaman detail penelitian, halaman data user, halaman input data user.

4.1.1 Halaman Awal Administrator

Pada tampilan halaman depan administrator terdiri dari link-link menu serta informasi jumlah data tanaman, penelitian, penyakit, cara pemakaian, bagian yang dapat digunakan sebagai obat, bagian tanaman. Tampilan dari halaman awal administrator adalah sebagai berikut:

The dashboard features a left-hand navigation menu with the following items: Dashboard, Tanaman, Penelitian, Khasiat, Bagian, Kegunaan, and Pemakaian. The main content area includes a 'Quote of the day' section with the text: "If you think you can, you really can". Below this are six informational cards: Informasi Tanaman, Informasi Penelitian, Informasi Khasiat, Informasi Bagian Tanaman, Informasi Kegunaan Tanaman, and Informasi Cara Pemakaian. At the bottom, there is a table titled 'Informasi web anda' showing various statistics.

Informasi web anda	
Jumlah Tanaman	50
Jumlah Penelitian	16
Jumlah Daftar Penyakit	83
Jumlah Daftar Cara Pemakaian	98
Jumlah Daftar Bagian Tanaman	16
Jumlah Daftar Bagian Yang di gunakan	116

Gambar 4.1: Tampilan depan halaman administrator

4.1.2 Halaman Data Tanaman Obat

Halaman data tanaman obat menampilkan data tanaman-tanaman yang mempunyai khasiat sebagai obat, halaman data tanaman obat terdiri dari kolom nomor, nama indonesia, nama daerah, nama asing, dan pilihan. Pada pilihan terdiri dari *icon* ubah yang berfungsi untuk mengubah data tanaman dengan cara menghubungkan data menuju form input data tanaman, *icon* hapus berfungsi untuk menghapus data tanaman obat dan *icon* detail untuk melihat detail tentang tanaman.

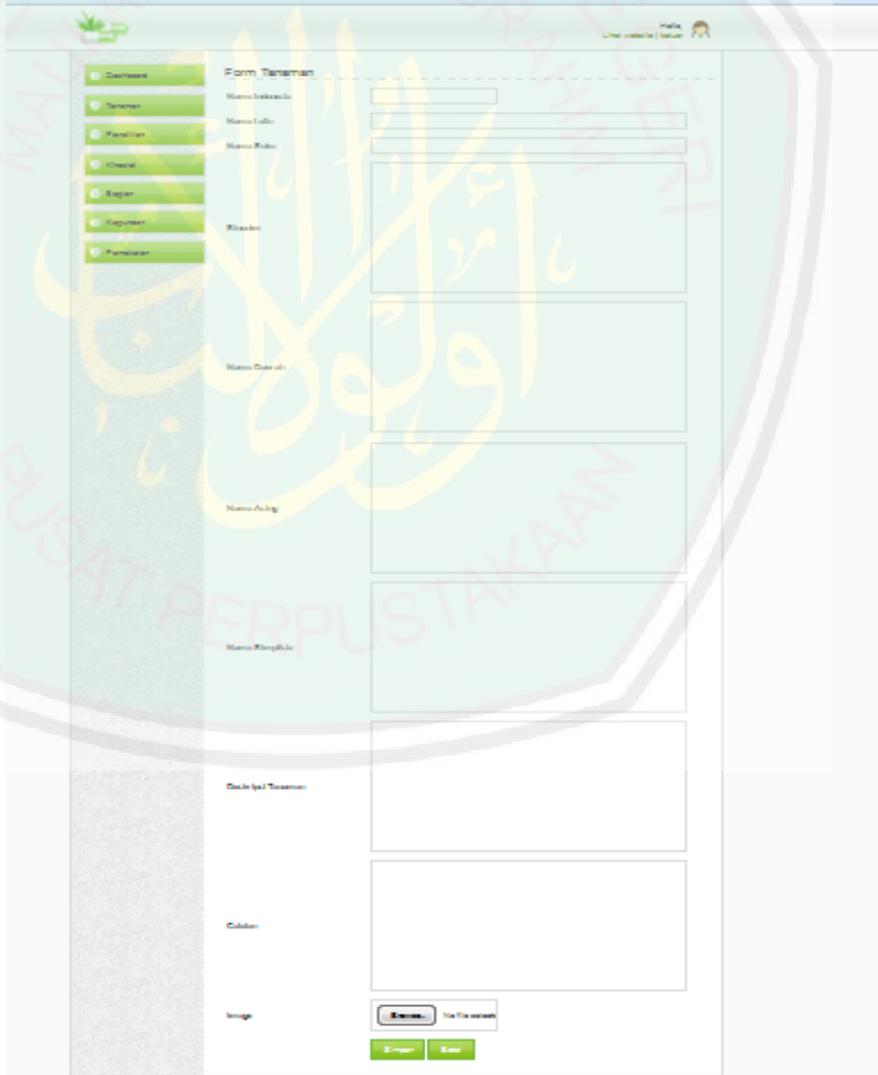
The 'Data Tanaman Obat' page displays a table with the following data:

No	Nama Indonesia	Nama Daerah	Nama Asing	Nama ilmiah	Pilihan
1	Sereh	Sumatra: sere mangat, sere, sange-sange, sarai, sorai. Jawa: sereh, sere. Kalimantan: serai, belanggak, sarai, sagumau. Nusa Tenggara: sée, pataha mpori, kendoung witi, nau sina, bu muke, lenian malai. Sulawesi: toti, timbu'ale, tangilo, tiwo mbane, sare, sere. Maluku: lapisa-pisa, hisa-hisa, hisa, isalo, bisa, bewuwu, gara ma fusu, biraema kusu, rimaniil	Xiag mao(C).	Cymbopogon nardus (L.) Rendle	
2	Semanggi Gunung	Pegagan embun, antana beuit, a lembut(Sunda), andem, kalepan, rendeng, semanggi(Jawa), Take cana(Madura), Salatun, tikim, patikim(Jakarta).	Tian hu sui(C), lawm pennywort(I).	Hydrocotyle sibthorpioides Lam.	
3	Selasih	Sumatra: selaseh, setasi. Jawa: solasih, selasih, telasih. Sulawesi: ampung, kururu.	Luo le(C), sweet basil, basil(I).	Ocimum basilicum L.	

Gambar 4.2: Tampilan data tanaman obat

4.1.3 Halaman *Input* Tanaman

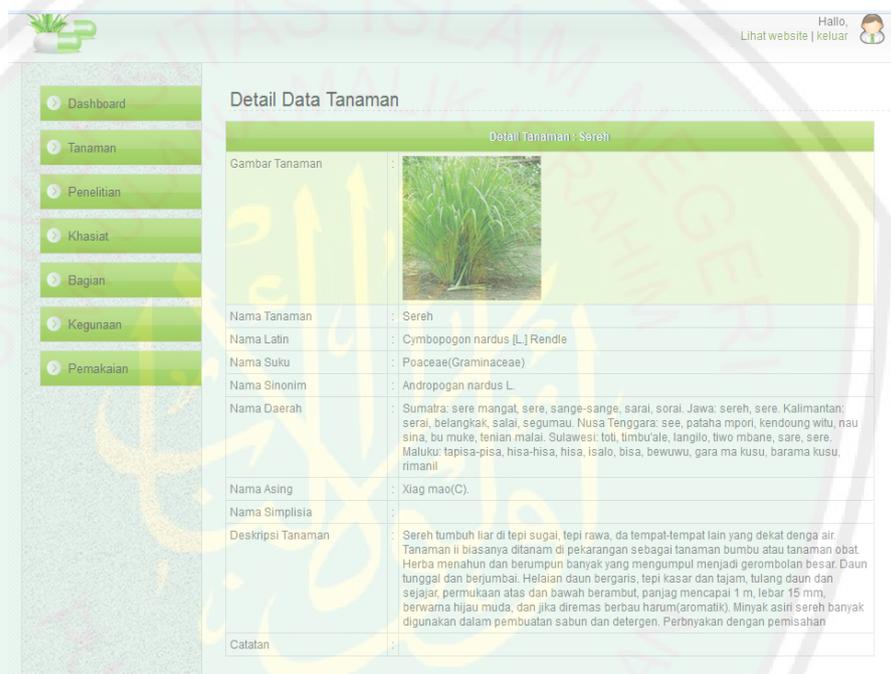
Halaman *input* tanaman berfungsi untuk menambah atau mengubah data tanaman obat. Halaman *input* tanaman terdiri dari *form* yang berisi nama Indonesia, nama latin, suku, sinonim, nama daerah, nama asing, nama simplisia, deskripsi tanaman, catatan, gambar tanaman, button “Batal” untuk membatalkan *input* data dan kembali ke halaman sebelumnya, button “Simpan” untuk menyimpan data serta button “Ubah” untuk mengubah data.



Gambar 4.3: Tampilan *input* tanaman

4.1.4 Halaman Detail Tanaman Obat

Pada halaman detail tanaman obat akan ditampilkan data lengkap dari tanaman yang terdiri dari gambar tanaman, nama tanaman, nama latin, nama suku, nama sinonim, nama daerah, nama asing, nama simplisia, deskripsi tanaman dan catatan.



Detail Data Tanaman	
Detail Tanaman : Sereh	
Gambar Tanaman	
Nama Tanaman	: Sereh
Nama Latin	: Cymbopogon nardus (L.) Rendle
Nama Suku	: Poaceae(Graminaceae)
Nama Sinonim	: Andropogon nardus L.
Nama Daerah	: Sumatra: sere mangat, sere, sange-sange, sarai, sorai. Jawa: sereh, sere. Kalimantan: serai, belangkak, salai, segumau. Nusa Tenggara: see, pataha mpori, kendoung witu, nau sina, bu muke, tenian malai. Sulawesi: toti, timbu'ale, langilo, tiwo mbane, sare, sere. Maluku: tapisa-pisa, hisa-hisa, hisa, isalo, bisa, bewuwu, gara ma kusu, barama kusu, rimanil
Nama Asing	: Xiag mao(C).
Nama Simplisia	:
Deskripsi Tanaman	: Sereh tumbuh liar di tepi sugai, tepi rawa, da tempat-tempat lain yang dekat denga air. Tanaman ii biasanya ditanam di pekarangan sebagai tanaman bumbu atau tanaman obat. Herba menahun dan berumpun banyak yang mengumpul menjadi gerombolan besar. Daun tunggal dan berjumbai. Helaiian daun bergaris, tepi kasar dan tajam, tulang daun dan sejajar, permukaan atas dan bawah berambut, panjang mencapai 1 m, lebar 15 mm, berwarna hijau muda, dan jika diremas berbau harumi(aromatik). Minyak asiri sereh banyak digunakan dalam pembuatan sabun dan detergen. Perbanyak dengan pemisahan
Catatan	:

Gambar 4.4: Tampilan Detail Tanaman Obat

4.1.5 Halaman Data Penelitian

Halaman Penelitian menampilkan data hasil penelitian tentang tanaman yang mempunyai khasiat sebagai obat. Halaman ini menampilkan data pada tabel yang terdiri dari nama tanaman, nama peneliti, tempat penelitian, dan pilihan. Pada kolom pilihan, terdapat *icon* ubah yang berfungsi untuk mengubah data penelitian dengan cara menghubungkan data menuju form input data penelitian,

icon hapus berfungsi untuk menghapus data penelitian dan *icon* detail untuk melihat detail tentang penelitian.

No.	Nama Tanaman	Nama Peneliti	Nama Tempat	Tahun	Pilihan
1	Saga	Hart M, et al	tidak tercantum	1963	
2	Pare	Bourinebariar	Biochem	1995	
3	Leunca	Konig H et al.	Chemical Abstracts	1953	
4	Salam	Retno Sudewi	Fakultas Farmasi UGM	1992	
5	Meniran	Elva Annisa	Fakultas Farmasi UGM	1991	
6	Temulawak	Abdul Naser	Jurusan Farmasi FMIPA UNPAD	1987	
7	Gandarusa	Hotma Elisa Siregar	Jurusan Farmasi FMIPA USU	1984	
8	Beluntas	Willys	Jurusan Farmasi FMIPA IAINAS	1990	

Gambar 4.5: Tampilan data penelitian

4.1.6 Halaman *Input* Penelitian

Halaman *input* penelitian berfungsi untuk menambah atau mengubah data penelitian tentang tanaman. Halaman *input* penelitian terdiri dari *form input* yang berisi nama peneliti, tempat penelitian, tahun penelitian, nama tanaman, hasil penelitian, button “Batal” untuk membatalkan input data dan kembali ke halaman sebelumnya, button “Simpan” untuk menyimpan data serta button “Ubah” untuk mengubah data.

Gambar 4.6: Tampilan *input* penelitian

4.1.7 Halaman Detail Penelitian

Pada halaman detail penelitian akan ditampilkan data lengkap tentang penelitian yang terdiri dari nama peneliti, tempat penelitian, tahun penelitian, nama tanaman dan hasil penelitian.



Gambar 4.7: Tampilan detail penelitian

4.1.8 Halaman Data Khasiat

Halaman data khasiat menampilkan data tentang khasiat tanaman obat pada tabel yang terdiri dari khasiat tanaman, dan pilihan. Pada pilihan terdapat *icon* ubah yang berfungsi untuk mengubah data khasiat dengan cara menghubungkan data menuju form input data tanaman, dan *icon* hapus berfungsi untuk menghapus data khasiat.

No	Khasiat Tanaman	Pilihan
1	TBC	
2	Mematangkan abses	
3	Batuk kering	
4	bercak putih	
5	Kanker	
6	Sifilis	
7	Disfungsi ereksi	
8	Digigit Ular	
9	Bengkak	

Gambar 4.8: Tampilan data khasiat

4.1.9 Halaman *Input* data Khasiat

Halaman *input* khasiat berfungsi untuk menambah atau mengubah data khasiat tentang tanaman. Halaman input khasiat terdiri dari *form* yang berisi nama penyakit, button “Batal” untuk membatalkan input data dan kembali ke halaman sebelumnya, button “Simpan” untuk menyimpan data serta button “Ubah” untuk mengubah data.



Gambar 4.9: Tampilan *input* khasiat

4.1.10 Halaman Data Bagian

Halaman data bagian menampilkan tabel berisi data bagian tanaman yang mempunyai khasiat sebagai obat. Pada kolom pilihan terdapat *icon* ubah yang berfungsi untuk mengubah data tanaman dengan cara menghubungkan data menuju form input data bagian tanaman dan *icon* hapus berfungsi untuk menghapus data bagian tanaman.

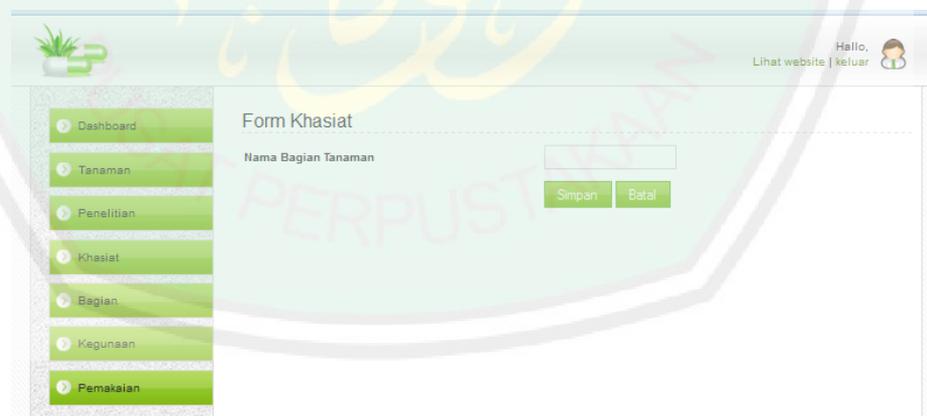


No	Nama Bagian Tanaman	Pilihan
1	Tangkai	
2	Getah	
3	Kayu	
4	tempurung	
5	air kelapa	
6	santan	
7	minyak	
8	kulit	

Gambar 4.10: Tampilan data bagian tanaman

4.1.11 Halaman *Input* Bagian

Halaman *input* bagian berfungsi untuk menambah atau mengubah data bagian tanaman. Halaman *input* bagian terdiri dari *form input* nama bagian tanaman, button “Batal” untuk membatalkan input data dan kembali ke halaman sebelumnya, dan button “Simpan” untuk menyimpan data serta button “Ubah” untuk mengubah data.



Form Khasiat

Nama Bagian Tanaman

Gambar 4.11: Tampilan *input* bagian

4.1.12 Halaman Kegunaan Bagian Tanaman

Halaman data kegunaan bagian tanaman obat menampilkan data kegunaan tanaman yang mempunyai kandungan kimia pada setiap bagiannya. Halaman data

kegunaan bagian tanaman terdiri dari kolom no, nama tanaman, nama bagian, kandungan kimia dan pilihan. Pada pilihan terdiri dari *icon* ubah yang berfungsi untuk mengubah data kegunaan bagian tanaman dengan cara menghubungkan data menuju form input data kegunaan bagian tanaman dan *icon* hapus berfungsi untuk menghapus data kegunaan bagian tanaman.



No	Nama Tanaman	Bagian Tanaman	Kandungan Kimia	Pilihan
1	Puring	Daun	tidak terdapat kandungan kimia	
2	Pulutan	Bunga	tidak terdapat kandungan kimia	
3	Nangka	Biji	tidak terdapat kandungan kimia	
4	Leunca	Daun	tidak terdapat kandungan kimia	
5	Sereh	Herba	Mengandung minyak asiri dengan komponen citronellal, citral, geraniol, methylheptenone, eugenol-methyl eter, dipentil, eugenol, kadinen, kadinol dan limonen.	
6	Semanggi Gunung	Herba	Mengandung minyak meguap, coumarin dan hyperin	
7	Selasih	Batang	tidak terdapat kandungan kimia	

Gambar 4.12 Tampilan data kegunaan bagian tanaman

4.1.13 Halaman *Input* Kegunaan Bagian Tanaman

Halaman *input* kegunaan bagian tanaman berfungsi untuk menambah atau mengubah data kegunaan bagian tanaman obat. Halaman input kegunaan bagian tanaman terdiri dari *form input* nama tanaman, bagian tanaman, kandungan kimia, button “Batal” untuk membatalkan input data dan kembali ke halaman sebelumnya, button “Simpan” untuk menyimpan data serta button “Ubah” untuk mengubah data.

Gambar 4.13 Tampilan *input* kegunaan bagian tanaman

4.1.14 Halaman Data Pemakaian

Halaman data pemakaian menampilkan informasi tentang cara pemakaian tanaman obat, halaman data pemakaian terdiri dari nama bagian, khasiat tanaman, cara pemakaian dan pilihan. Pada kolom pilihan terdapat *icon* ubah yang berfungsi untuk mengubah data cara pemakaian dengan cara menghubungkan data menuju form input data pemakaian dan *icon* hapus berfungsi untuk menghapus data cara pemakaian.

No	Nama Tanaman	Bagian Tanaman	Khasiat Tanaman	Cara Pemakaian	Pilihan
1	Wortel	Akar	Beri-beri	Rebus 2 sendok makan bunga wortel kering dalam 3 gelas air sampai tersisa 1 gelas. Setelah dingin, saring dan air saringannya diminum sekaligus. Lakukan 2 kali sehari, pagi dan siang hari.	
2	Temulawak	Rimpang	hepatitis	1. Rimpang temulawak segar sebesar 2 jari dikupas kulitnya lalu dipang. Tambahkan air panas sebanyak 1/2 cangkir dan 1 sendok makan madu. Aduk campuran tadi sampai merata lalu diartikan mengendap. Minum beningannya, ampasnya dibuang. Lakukan 2 kali sehari, sampai sembuh. 2. Rebus 10 g rimpang temulawak kering dan 30 g akar alang alang (<i>Imperata cylindrica</i>) dalam 3 gelas air sampai tersisa 1 gelas. Setelah dingin, saring dan air saringannya diminum sekaligus. Lakukan 2 kali sehari.	
3	Sereh	Herba	nyeri lambung	Rebus sebanyak 5-15 g atau ambil secukupnya, lalu rendam dengan arak. Bisa juga dengan meminum beberapa tetes minyak sereh.	

Gambar 4.14 Tampilan data cara pemakaian tanaman

4.1.15 Halaman *Input* Pemakaian

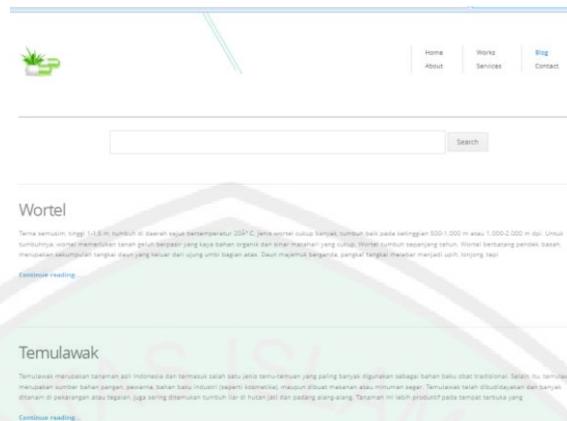
Halaman input pemakaian berfungsi untuk menambah atau mengubah data cara pemakaian tanaman obat. Halaman input bagian tanaman terdiri dari *form input* nama bagian, khasiat tanaman, cara pemakaian, button “Batal” untuk membatalkan input data dan kembali ke halaman sebelumnya, button “Simpan” untuk menyimpan data serta button “Ubah” untuk mengubah data.

Gambar 4.15 Tampilan input cara pemakaian

4.2 *Preprocessing* Pencarian

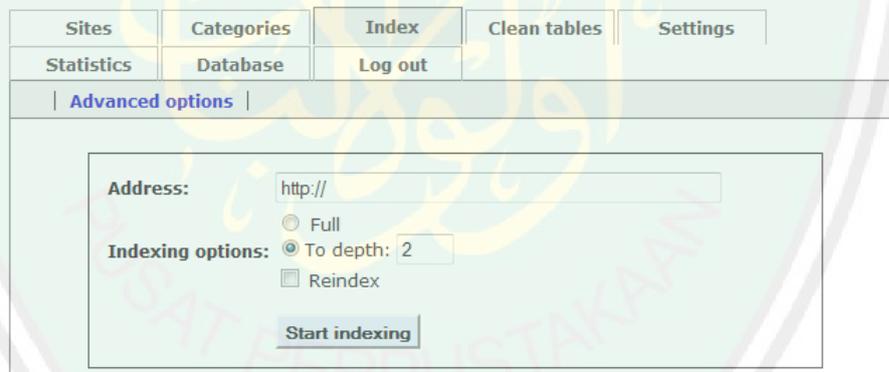
4.2.1 *Crawling*

Proses *crawling* yaitu membaca semua data url dari <http://tobat.web-semantik.com>. Proses *Crawling* pada penelitian ini dilakukan dengan menggunakan aplikasi sphider (Sphider, 2013). *Crawling* ditujukan untuk *download* semua data url yang terdapat pada domain aplikasi kemudian *memparsing* dan menyimpannya kedalam *repository crawling data*.



Gambar 4.16 Tampilan <http://tobat.web-semantik.com>

Aplikasi spider akan membaca link terkait dari halaman (Address) awal yang di masukkan hingga kedalaman yang ingin dicari (To depth). Pada aplikasi spider ini berfungsi untuk meng*crawl* semua link yang terdapat pada tab Index.



Gambar 4.17 Tampilan aplikasi Spider untuk untuk proses *Crawling*

Ketika tombol 'Start indexing' dijalankan maka aplikasi ini akan melakukan proses membaca semua data url, *mendownload* dan menyimpan url-ur tersebut. Hasil dari proses *crawling* tanaman obat sebagai berikut:

Gambar 4.18 Tampilan Hasil *Crawling*

4.2.2 Indexing

Proses *indexing* pada penelitian ini menggunakan function php `simple_html_domp.php` (sourcefoge, 2013). Function `simple_html_domp.php` digunakan untuk membaca tag yang terdapat dalam *source code* html. Pada penelitian ini function `simple_html_domp.php` menggunakan *parsing title* dan *content* pada *tag body*.

```

. . .
require 'simple_html_dom.php';
$query_data = mysql_query("SELECT link FROM temp ORDER BY link ");
while($data = mysql_fetch_array($query_data)){
    $isi = $data['link'];
    $html = file_get_html($isi);
    foreach($html->find('body') as $article) {
        $title = $article->find('div.title', 0)->plaintext; echo'<br>';
    }
    foreach($html->find('body') as $article) {
        $text= $article->find('div.deskripsi', 0)->plaintext; echo'<br>';
    }
}
. . .

```

Gambar 4.19 *Source Code Parsing title* dan *content* pada *tag body*

Proses selanjutnya setelah data di-*parsing* adalah pencarian serta perhitungan kata yang berhubungan dengan tanaman obat pada *title* dan *content* serta menyimpannya dalam tabel indexing.

```

. . . .
$query_data2 = mysql_query("SELECT * FROM word ORDER BY word ASC ");
while($data2 = mysql_fetch_array($query_data2)){
    $word = $data2['word'];
    $count_title = count(explode($word, $title))-1;
    $count_body = count(explode($word, $text))-1;

    mysql_query("INSERT INTO indexing(id_index, link, word , count_title,
    count_content) VALUES ('1', '$isi', '$word', '$count_title', '$count_body')");
}
. . . .

```

Gambar 4.20 *Source* perhitungan kata dan *insert* kedalam tabel indexing



Gambar 4.21 *Interface* Proses Indexing

4.2.3 Pembuatan Metadata

Pembuatan metadata dilakukan dengan menggunakan aplikasi Smore dengan menggunakan data ontologi berformat .owl yang telah dibuat sebelumnya dengan menggunakan aplikasi Protégé. Metadata pada setiap url berbentuk *triple table* yaitu terdiri dari subject, predikate dan object.

Subject	Predicate	Object
<test:daun_alamanda>	<rdfs:type>	<owl:Thing>
<test:daun_alamanda>	<Ontology1365459030:bagian_dari>	<Ontology1365459030:Saga>
<test:daun_alamanda>	<Ontology1365459030:digunakan_untuk>	<Ontology1365459030:buah_kelapa>
<test:daun_alamanda>	<Ontology1365459030:dipakai_dengan>	<test:leunca_daun>
<test:daun_alamanda>	<Ontology1365459030:diteliti_oleh>	<Ontology1365459030:Kembang_Telang>
<test:daun_alamanda>	<Ontology1365459030:kandungan_kimia_dari>	<Ontology1365459030:kayu>
<test:daun_alamanda>	<Ontology1365459030:mempunyai_kandungan_kimia>	<Ontology1365459030:Saga>
<test:daun_alamanda>	<Ontology1365459030:mempunyai_bagian>	<Ontology1365459030:kayu>
<test:daun_alamanda>	<Ontology1365459030:mempunyai_nama_asing>	<test:leunca_daun>
<test:daun_alamanda>	<Ontology1365459030:mempunyai_nama_daerah>	<Ontology1365459030:biji_nangka>
<test:daun_alamanda>	<Ontology1365459030:mempunyai_nama_latin>	<Ontology1365459030:diare>
<test:daun_alamanda>	<Ontology1365459030:mempunyai_nama_simplisia>	<Ontology1365459030:Saga>
<test:daun_alamanda>	<Ontology1365459030:mempunyai_sinonim>	<Ontology1365459030:Bunga_mengandung_quercime...>
<test:daun_alamanda>	<Ontology1365459030:mempunyai_suku>	<Ontology1365459030:buah_patikan_kebo>
<test:daun_alamanda>	<Ontology1365459030:nama_asing_dari>	<http://www.owl-ontologies.com/Ontology1365459030.owl#bunga_patikan_kebo>
<test:daun_alamanda>	<Ontology1365459030:nama_daerah_dari>	<Ontology1365459030:diare>
<test:daun_alamanda>	<Ontology1365459030:nama_latin_dari>	<Ontology1365459030:Xiang_si_ji_C_wild_liquorice_in...>
<test:daun_alamanda>	<Ontology1365459030:nama_simplisia_dari>	<Ontology1365459030:Asteraceae_Compositae>
<test:daun_alamanda>	<Ontology1365459030:suku_dari>	<Ontology1365459030:buah_nangka>
<test:daun_alamanda>	<Ontology1365459030:pemakaian_dari>	<test:karet_kebo_akar>
<test:daun_alamanda>	<Ontology1365459030:penelitian_dari>	<test:leunca_daun>
<test:daun_alamanda>	<Ontology1365459030:penggunaan_dari>	<Ontology1365459030:Sambilotto>
<test:daun_alamanda>	<Ontology1365459030:sinonim_dari>	<Ontology1365459030:diare>

Gambar 4.22 Triple Table hasil pemberian metadata

```

...
<?xml version="1.0"?>
<!DOCTYPE owl [
  <!ENTITY owl "http://www.w3.org/2002/07/owl#">
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#">
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#">
]>
<rdf:RDF
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:Ontology1365459030="http://www.owl-ontologies.com/Ontology1365459030.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:test="file:/C:/MINDlab/test.owl#"
>
  <owl:Ontology rdf:about="file:/C:/MINDlab/test.owl">
  </owl:Ontology>
  <owl:Class rdf:about="http://www.w3.org/2002/07/owl#Thing">
  </owl:Class>
  <owl:ObjectProperty
    ontology1365459030.owl#bagian_dari">
  </owl:ObjectProperty>
  <owl:ObjectProperty
    ontology1365459030.owl#digunakan_untuk">
  </owl:ObjectProperty>
  <owl:ObjectProperty
    ontology1365459030.owl#dipakai_dengan">
  </owl:ObjectProperty>
  </owl:RDF>

```

Gambar 4.23 Hasil Metadata dengan Menggunakan Smore

Triple table yang tersimpan dalam format .owl akan diparsing untuk menentukan Subjek, Predikat dan Objek kemudian disimpan ke dalam database MySQL pada tabel ntriple menggunakan bantuan *library* API JENA.

```

. . .
Model model = ModelFactory.createDefaultModel();
String inputFileNames = file.toString();
System.out.println(inputFileNames);
InputStream in = FileManager.get().open(inputFileNames);
model.read(in, "");

String data = model.write( System.out, "N-TRIPLE").toString();
txresult.append(data);
. . .

```

Gambar 4.24 Source code untuk membaca N-Triple

```

. . .
StringTokenizer st = new StringTokenizer(data, "[ ]" );
st.nextToken();
while(st.hasMoreTokens()){
String data2 = st.nextToken();
StringTokenizer st2 = new StringTokenizer(data2, ", <>");
while(st2.hasMoreTokens()){
String S = st2.nextToken();
String P = st2.nextToken();
String O = st2.nextToken();
}
}
. . .

```

Gambar 4.25 Source code parsing SPO

Subjek	Predikat	Objek	Link
http://www.owl-ontologies.com/Ontology13654590...	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing	http://localhosttanaman/Folium_Cinkgo.html
http://www.owl-ontologies.com/Ontology13654590...	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing	http://localhosttanaman/radang.html
http://www.owl-ontologies.com/Ontology13654590...	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing	http://localhosttanaman/buah.html
http://www.owl-ontologies.com/Ontology13654590...	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing	http://localhosttanaman/Sambiloto.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.owl-ontologies.com/Ontology13654590...	http://www.owl-ontologies.com/Ontology13654590...	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing	http://localhosttanaman/puring.html
file:/C:/MINDIab/test/owl/puring_akar	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing	http://localhosttanaman/akar_karet.html
http://www.owl-ontologies.com/Ontology13654590...	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing	http://localhosttanaman/Capparidaceae.html
http://www.owl-ontologies.com/Ontology13654590...	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing	http://localhosttanaman/Mamang.html
http://www.owl-ontologies.com/Ontology13654590...	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing	http://localhosttanaman/rasin.html

Gambar 4.26 Tampilan aplikasi untuk membaca dan memasukkan SPO kedalam database

4.3 Proses Pencarian

Proses pencarian semantik dalam aplikasi ini dilakukan melalui pencarian terhadap *repository indexing* dan *repository n-triple*, dimana dari hasil pencarian dari kedua proses tersebut kemudian dicocokkan dengan menggunakan Algoritma Boyer-Moore. Proses Pencarian dalam penelitian ini dibagi kedalam beberapa tahapan sebagai berikut:

4.3.1 Menyaring Kata Penting Dari Kalimat Pencarian

Kalimat pencarian yang dimasukkan oleh user akan disaring untuk menentukan kata-kata yang berhubungan dengan tanaman obat. Setiap kata akan dicocokkan dengan kata yang berhubungan dengan tanaman obat dari tabel word yang terdapat pada *database*. Potongan *source code* pada gambar 4.27 menunjukkan proses ketika *user* memasukkan kalimat pencarian maka sistem akan menyaring kata yang dianggap penting dan membuang kata penghubung yang telah ditabelkan sebelumnya pada tabel 'kata_penghubung'.

```

. . .
$search = $_GET['key'];
$tok = strtok($search, "_");
while ($tok != false) {
    $query_data = mysql_query("SELECT * FROM word WHERE word = '$tok'");
    $data = mysql_fetch_array($query_data);

    $query_data2 = mysql_query("SELECT * FROM kata_penghubung WHERE
kata_penghubung LIKE '%$tok%'");

    $data2 = mysql_fetch_array($query_data2);

    if($data==null){
    }else if($data2!=null){
    }else{
        $array[$j] = "$tok";
        $j++;
    }
    $low = Strtolower($tok);
    $tok = strtok("_");
    $i++;
}
. . .

```

Gambar 4.27 *Source code* penyaringan kata

4.3.2 Pencarian Kata Pada Tabel *Indexing*

Dari hasil *indexing*, kata yang telah disaring selanjutnya dicari pada tabel *indexing* untuk menentukan urutan keterkaitan artikel dengan kata yang dicari. Pencocokan kata pada *repository indexing* kemudian disimpan ke dalam *array_indexing*. *Repository indexing* ini menyediakan jumlah kata penting yang dicari dalam *title* dan *body* pada masing-masing link dokumen, sehingga hasil dari pencarian ini diperoleh data dengan ranking tertinggi sampai dengan terendah sesuai dengan jumlah kata penting yang ditemukan dalam tag *title* dan *body*.

```

. . .
$counter = count($array);
for($a=2; $a<=$counter; $a++){
    $sql_indexing = $sql_indexing." OR word LIKE '%$array[$k]%'";
}

$sql_indexing = $sql_indexing." ORDER BY count_title DESC, count_content DESC";
$query_indexing = mysql_query($sql_indexing);
while($indexing = mysql_fetch_array($query_indexing)){
    $array_indexing[$k] = $indexing['link'];
    $k++;
}
. . .

```

Gambar 4.28 *Source code* pencocokan kata pada tabel *indexing*

4.3.3 Pencarian Kata Pada Tabel *Ntriple*

Proses ini dilakukan dengan tujuan mendapatkan keakuratan data dengan pencariannya terhadap metadata yang telah disimpan sebelumnya pada tabel ‘*ntriple*’ (gambar 4.29). Kata yang telah disaring selanjutnya dicari pada tabel ‘*ntriple*’ untuk menentukan keterkaitan kalimat dengan objek yang ada pada tabel ‘*ntriple*’.

```

. . .
for($a=1; $a<=$counter; $a++){
    $sql_ntriple = "SELECT DISTINCT(link) FROM ntriple WHERE objek LIKE
'%" . $array[$a] . "%'";
    $query_ntriple = mysql_query($sql_ntriple);

    while($ntriple = mysql_fetch_array($query_ntriple)){
        $array_ntriple2[$l] = $ntriple['link'];
        $l++;
    }
}
$counter_ntriple2 = count($array_ntriple2);

for($b=1; $b<=$counter_ntriple2; $b++){
    for($c=1; $c<=$counter_ntriple2; $c++){
        if($array_ntriple2[$b]==$array_ntriple2[$c]){
            $jum = $jum+1;
            if($jum == $counter){
                $array_ntriple[$m] = $array_ntriple2[$c];
                $m++;
            }
        }
    }
}
$jum = 0;
}
. . .

```

Gambar 4.29 *Source code* pencocokan kata pada tabel ntriple

4.3.4 Pencocokan Kata Hasil *Indexing* Dengan Kata Hasil *Ntriple* Menggunakan Metode Booyer-Moore

Hasil dari pencocokkan kata dengan tabel indexing dan pencocokkan kata dengan tabel ntriple selanjutnya akan dicocokkan dengan Metode Booyer-Moore sebagai algoritma *string matching*nya untuk mendapatkan hasil yang akurat. Gambar 4.30 merupakan potongan *source code* untuk proses ini.

```

. . .
include ("boyer-more2.php");
for($b=1; $b<=$counter_indexing; $b++){
    for($c=1; $c<=$counter_ntriple3; $c++){
        boyer_moore($array_ntriple3[$c], $array_indexing[$b], $cv);
    }
}
. . .

```

Gambar 4.30 *Source code* pencocokan hasil indexing dan ntriple dengan metode Boyer-

Moore

4.3.5 Merubah hasil pencarian ke dalam bentuk J-SON

Hasil dari pencarian dirubah menjadi bentuk JSON agar dapat diakses menggunakan perangkat *mobile*.

```

. . .
$link = substr($text,25,50);
$url = substr($text,25,strpos($link, "."));
$q = mysql_query("SELECT * FROM tanaman WHERE nama_url = ".$url."");

$r=mysql_fetch_array($q);
$ob = array('','<br>','</br>');
$cut = substr($r["deskripsi_tanaman"],0,100);
$desc = substr($r["deskripsi_tanaman"],0, strpos($cut, " "));
$v = '{"id" : "'.$r['idtanaman'].'", "name" : "'.str_replace($ob,'
',strip_tags($r["nama_indo"])).'", "desc" : "'.str_replace($ob,'
',strip_tags($desc)).'"}';

echo $v;
. . .

```

Gambar 4.31: *Source code* merubah hasil menjadi bentuk J-SON

4.4 Proses Pencarian Menggunakan Metode Boyer-Moore

Algoritma Boyer-Moore dalam aplikasi ini digunakan untuk mencocokkan hasil dari pencarian pada tabel indexing dan hasil pencarian dari tabel ntriple. Sebelum proses pencarian dimulai akan dilakukan perhitungan nilai dari tabel *Bad Characters* dan *Good Suffix*. Selanjutnya dalam proses pencarian pergeseran dilakukan sesuai kesalahan yang terjadi dan nilai dari tabel.

4.4.1 Perhitungan *Bad Character*

Bad Characters dihitung dari *string pattern* dengan cara memasukkan posisi tertinggi semua karakter alfabet pada *pattern*. Bila *pattern* tidak mengandung karakter tersebut maka nilainya adalah -1, yang nanti akan menyebabkan pergeseran sebesar panjang *pattern* tersebut.

```

. . .
function bad_characters($pattern, &$bad_chars)
{
    $patern_l = strlen($pattern);

    for ($i = 0; $i < $patern_l - 1; ++$i) {
        $bad_chars[$pattern{$i}] = $patern_l - $i - 1;
    }
}
. . .

```

Gambar 4.32: Source code perhitungan bad character

4.4.2 Perhitungan Good Suffix

Good Suffix dihitung dari *pattern* yang ada pada karakter saat verifikasi salah. Dua kasus yang terjadi adalah *pattern* yang cocok terjadi juga di *string pattern* dan hanya sebagian dari *pattern* yang cocok ada pada awal dari *string pattern*. Dari kedua kasus tersebut nilai yang paling besar diambil untuk setiap *pattern* karakter dari *i* sampai panjang *string pattern*.

```

. . .
function good_suffixes($pattern, &$good_suffixes){
    $patern_l = strlen($pattern);
    $suff = array();

    suffixes($pattern, $suff);

    for ($i = 0; $i < $patern_l; $i++) {
        $good_suffixes[$i] = $patern_l;
    }

    for ($i = $patern_l - 1; $i >= 0; $i--) {
        if ($suff[$i] == $i + 1) {
            for ($j = 0; $j < $patern_l - $i - 1; $j++) {
                if ($good_suffixes[$j] == $patern_l) {
                    $good_suffixes[$j] = $patern_l - $i - 1;
                }
            }
        }
    }

    for ($i = 0; $i < $patern_l - 2; $i++) {
        $good_suffixes[$patern_l - 1 - $suff[$i]] = $patern_l - $i - 1;
    }
}
. . .

```

Gambar 4.33: Source code perhitungan good suffix

4.5 Implementasi *Interface* Pada *Mobile*

4.5.1 Halaman Awal Pada Aplikasi

User interface aplikasi ensiklopedia tanaman obat tradisional berbasis *mobile* ini memiliki 4 menu di halaman utamanya yaitu menu pencarian, menu data tanaman A-Z, menu informasi dan menu keluar. Pada saat aplikasi dijalankan sistem akan menampilkan halaman awal seperti yang terdapat pada gambar 4.34.



Gambar 4.34 *Interface* awal user

4.5.2 *Interface* List Pencarian

Pada *interface* ini akan memberikan informasi nama tanaman beserta manfaatnya dari A-Z. Jika ingin mengetahui nama tanaman obat yang berhuruf A maka mengklik list A dan tanaman yang memiliki nama depan A akan muncul. Kemudian jika ingin mengetahui informasi nama tanaman tersebut maka user dapat mnegklik nama tanaman maka akan muncul informasi nama tanaman sesuai yang dipilih oleh user.



Gambar 4.35 *Interface* list semua data tanaman obat

4.5.3 *Interface* Pencarian

Merupakan menu yang menyediakan mesin pencarian bagi *user* (gambar 4.36). Disini *user* dapat melakukan pencarian dengan memasukkan kalimat pencarian yang selanjutnya diproses oleh sistem untuk menampilkan hasil. Pada halaman ini user dapat menginputkan kata yang berkaitan dengan tanaman obat kemudian menekan tombol cari maka akan ditampilkan informasi tanaman obat yang dicari.



Gambar 4.36 *Interface* pencarian tanaman obat

4.5.4 Interface Hasil dari Pencarian

Interface hasil pencarian menampilkan data hasil pencarian yang telah diinputkan oleh user.



Gambar 4.37 Interface hasil pencarian

4.5.5 Detail Hasil Pencarian

Pada halaman ini akan di tampilkan secara lebih detail tentang tanaman obat yang ada.



Gambar 4.38 Tampilan detail pencarian tanaman obat

4.6 Uji Coba Aplikasi

Kriteria untuk pengujian performa dari teknik pencarian yaitu dengan mengukur waktu pencarian, keakuratan pencarian dan memori yang diperlukan. Proses pengujian dalam penelitian ini yaitu melakukan pengukuran kecepatan pencarian dan keakuratan hasil pencarian.

Uji coba dilakukan dengan menggunakan *mobile* samsung S5360 dengan sistem operasi Android 2.3 (Gingerbread) dan menggunakan jaringan HDSPA dengan hasil sebagai berikut :

Tabel 4.1 Uji coba perbandingan kecepatan

No	Kalimat Pencarian	Waktu yang Dibutuhkan (s)			
		T OBAT	Wordpress	Joomla	Drupal
1	Tanaman yang dapat digunakan untuk menyembuhkan batuk	01.30	02.70	01.65	01.73
2	Khasiat dari wortel	01.40	02.90	02.12	02.91
3	Tanaman yang mengandung Alonin	01.10	02.10	01.50	02.24
4	Tanaman yang dapat mengobati rematik dengan diminum	01.70	02.50	01.68	02.70
5	Tanaman obat yang terdapat dalam suku zingiberaceae	01.30	03.10	01.60	02.58
6	Curcuma Rhizoma	01.00	02.70	01.28	01.84
7	Tanaman yang bisa digunakan untuk menyembuhkan gatal	00.70	03.20	01.35	02.43
8	Tanaman yang dapat digunakan untuk menyembuhkan batuk	01.00	02.20	01.37	01.60
9	Tanaman obat yang terdapat dalam suku poaceae	01.20	03.20	01.64	02.66
10	Tanaman obat yang memiliki nama latin cymbopogon	01.10	02.70	01.83	01.25
11	Tanaman yang dapat digunakan untuk mengobati kencing manis	01.00	03.30	02.50	01.34
12	Minyak asiri terdapat pada tanaman	01.50	03.20	01.15	01.72
13	Xiang si ji merupakan nama asing dari tanaman	01.90	03.10	01.32	01.43
14	Khasiat dari tanaman saga	01.20	02.20	01.51	01.45
15	Tanaman yang dapat menyembuhkan Bisul	01.70	02.90	01.12	02.51

16	Legundi dapat digunakan untuk menyembuhkan?	01.50	02.00	01.41	01.34
17	Tanaman obat yang memiliki kandungan saponin	01.60	02.00	01.53	01.55
18	Tanaman obat yang dapat digunakan untuk menyembuhkan disentri	01.20	03.20	01.30	01.74
19	Tanaman obat yang dapat digunakan untuk menyembuhkan demam	01.50	02.80	01.19	01.43
20	Alamanda	01.40	03.10	01.63	02.28
21	Nangka	01.70	02.70	01.55	01.91
22	Saga	01.70	02.30	01.49	02.13
23	Koolboom	01.00	03.10	01.71	02.21
24	Jing feng hua	01.10	02.70	01.21	02.41
25	Cocos Nufera	01.30	02.30	01.40	01.64

Tabel 4.2 Uji coba perbandingan keakuratan

No	Kalimat Pencarian	Hasil Pencarian			
		T OBAT	Wordpress	Joomla	Drupal
1	Tanaman yang dapat digunakan untuk menyembuhkan batuk	5 Tepat	Tidak ada hasil	Tidak ada hasil	Tidak ada hasil
2	Khasiat dari wortel	1 Tepat	1 Tepat	1 Tepat	1 Tepat
3	Tanaman yang mengandung Alonin	1 Tepat	1 Tepat	Tidak ada hasil	1 Tepat
4	Tanaman yang dapat mengobati rematik dengan diminum	3 Tepat 1 Salah	Tidak ada hasil	Tidak ada hasil	Tidak ada hasil
5	Tanaman obat yang terdapat dalam suku zingiberaceae	2 Tepat	2 Tepat	Tidak ada hasil	2 Tepat
6	Curcuma Rhizoma	1 Tepat	1 Tepat	1 Tepat	1 Tepat
7	Tanaman yang bisa digunakan untuk menyembuhkan gatal	3 Tepat	3 Tepat	Tidak ada hasil	Tidak ada hasil
8	Tanaman yang dapat digunakan untuk menyembuhkan batuk	2 Tepat	Tidak ada hasil	Tidak ada hasil	Tidak ada hasil
9	Tanaman obat yang terdapat dalam suku poaceae	2 Tepat	1 Tepat	Tidak ada hasil	1 Tepat
10	Tanaman obat yang memiliki nama latin cymbopogon	1 Tepat	Tidak ada hasil	Tidak ada hasil	Tidak ada Hasil
11	Tanaman yang dapat	1Tepat	Tidak ada	Tidak ada	Tidak ada

	digunakan untuk mengobati kencing manis		hasil	hasil	Hasil
12	Minyak asiri terdapat pada tanaman	1 Tepat	1 Tepat	Tidak ada hasil	1 Tepat
13	Xiang si ji merupakan nama asing dari tanaman	1 Tepat 1 Salah	1 Salah	Tidak ada hasil	Tidak ada hasil
14	Khasiat dari tanaman saga	1 Tepat	1 Tepat	Tidak ada hasil	1 Tepat
15	Tanaman yang dapat menyembuhkan Bisul	2 Tepat	1 Tepat	Tidak ada hasil	Tidak ada hasil
16	Legundi dapat digunakan untuk menyembuhkan?	1 Tepat	Tidak ada hasil	Tidak ada hasil	Tidak ada hasil
17	Tanaman obat yang memiliki kandungan saponin	1 Tepat	1 Tepat	Tidak ada hasil	Tidak ada hasil
18	Tanaman obat yang dapat digunakan untuk menyembuhkan disentri	4 Tepat	1 Tepat	Tidak ada hasil	Tidak ada hasil
19	Tanaman obat yang dapat digunakan untuk menyembuhkan demam	1 Tepat	1 Tepat	Tidak ada hasil	Tidak ada hasil
20	Alamanda	1 Tepat	1 Tepat	1 Tepat	1 Tepat
21	Nangka	1 Tepat	1 Tepat 2 Salah	1 Tepat 2 Salah	1 Tepat
22	Saga	1 Tepat	1 Tepat	1 Tepat	1 Tepat
23	Koolboom	1 Tepat	1 Tepat	1 Tepat	1 Tepat
24	Jing feng hua	2 Tepat	2 Tepat 8 Salah	1 Tepat	Tidak ada hasil
25	Cocos Nufera	1Tepat	1 Tepat	Tidak ada hasil	1 Tepat

Tabel 4.3: Perbandingan keakuratan dan kecepatan rata-rata

Nama Aplikasi	Kecepatan	Keakuratan
T OBAT	01.32	$39/41 * 100\% = 95\%$
Wordpress	02.28	$22/41 * 100\% = 53\%$
Joomla	01.52	$5/41 * 100\% = 12\%$
Drupal	01.96	$13/41 * 100\% = 31\%$

Dari hasil uji coba yang telah dilakukan dapat diketahui bahwa keakuratan aplikasi pada penelitian ini mencapai 95% dengan kecepatan rata-rata 01.32 s,

keakuratan wordpress 53% dengan kecepatan rata-rata 02.28, keakuratan joomla 12% dengan kecepatan rata-rata 01.52 dan keakuratan drupal 31% dengan kecepatan rata-rata 01.96

Aplikasi pada penelitian ini memberikan hasil informasi tentang tanaman obat meliputi : nama indonesia, nama daerah, nama latin, nama asing, nama simplisia, deskripsi, manfaat, cara penggunaan serta penelitian dari tanaman obat.

4.7 Integrasi Keilmuan

Indonesia memiliki kekayaan alam berupa aneka ragam jenis tanaman dengan berbagai manfaat. Dari berbagai jenis tanaman yang ada di Indonesia sebagian besar diantaranya dapat dimanfaatkan untuk obat tradisional. Allah SWT telah menciptakan tumbuh-tumbuhan secara melimpah dengan banyak manfaatnya bagi umat manusia, sebagaimana dijelaskan dalam Al-Qur'an surat Thaahaa ayat 53 sebagai berikut:

الَّذِي جَعَلَ لَكُمُ الْأَرْضَ مَهْدًا وَسَلَكَ لَكُمْ فِيهَا سُبُلًا وَأَنْزَلَ مِنَ السَّمَاءِ
مَاءً فَأَخْرَجْنَا بِهِ أَزْوَاجًا مِّن نَّبَاتٍ شَتَّى ﴿٥٣﴾

“Yang telah menjadikan bagimu bumi sebagai hamparan dan Yang telah menjadikan bagimu di bumi itu jalan-jalan, dan menurunkan dari langit air hujan. Maka Kami tumbuhkan dengan air hujan itu berjenis-jenis dari tumbuh-tumbuhan yang bermacam-macam.” (QS. Thaahaa : ayat 53)”

Tanaman yang dapat dimanfaatkan sebagai obat memiliki khasiat yang beragam untuk menyembuhkan berbagai penyakit. Obat-obatan yang berasal dari tumbuhan sangat baik karena lebih bersifat alami dan tidak menggunakan bahan

kimia jika dibandingkan dengan obat-obatan saat ini yang banyak bersifat anorganik dan cenderung mengandung bahan kimia. Obat herbal sangat bermanfaat, lebih terjangkau dan aman karena memiliki efek samping yang relatif kecil jika digunakan dengan tepat, baik takaran, waktu dan cara penggunaan, pemilihan bahan serta penyesuaian dengan indikasi tertentu.

Rasulullah SAW menganjurkan melalui hadits berikut :

“Sesungguhnya Allah menurunkan penyakit dan obat, dan menjadikan bagi setiap penyakit obatnya, maka (berobatlah kamu sekalian, tetapi) jangan berobat dengan yang haram.” (HR. Abu Dawud)”.

Dari berbagai macam tanaman obat yang ada, banyak manfaat dari tanaman-tanaman obat ini yang hanya diketahui oleh sebagian orang dikarenakan kurangnya pengetahuan dan informasi yang mendukung. Aplikasi *mobile* ensiklopedia tanaman obat tradisional yang telah diperoleh dari hasil penelitian ini, diharapkan dapat lebih memudahkan masyarakat dalam memperoleh informasi tanaman obat melalui perangkat *mobile* sehingga lebih efektif dan efisien dalam penggunaannya.

Pencarian informasi sebenarnya telah diriwayatkan dalam hadits Abu Dawud no.3641, bahwa Abu Darda' *radhiyallahu 'anhu* berkata : aku mendengar Rasulullah *shallahu 'alaihi wa salam* bersabda :

مَنْ سَلَكَ طَرِيقًا يَطْلُبُ فِيهِ عِلْمًا سَلَكَ اللَّهُ بِهِ طَرِيقًا مِنْ طُرُقِ الْجَنَّةِ وَإِنَّ الْمَلَائِكَةَ لَتَضَعُ أَجْنِحَتَهَا رِضًا لِطَالِبِ الْعِلْمِ وَإِنَّ الْعَالِمَ لَيَسْتَغْفِرُ لَهُ مَنْ فِي السَّمَوَاتِ وَمَنْ فِي الْأَرْضِ وَالْحَيَّاتَانِ فِي جَوْفِ الْمَاءِ وَإِنَّ فَضْلَ الْعَالِمِ عَلَى الْعَابِدِ كَفَضْلِ الْقَمَرِ لَيْلَةً

الْبَدْرِ عَلَى سَائِرِ الْكَوَاكِبِ وَإِنَّ الْعُلَمَاءَ وَرَثَةُ الْأَنْبِيَاءِ وَإِنَّ الْأَنْبِيَاءَ لَمْ يُورَثُوا دِينَارًا وَلَا دِرْهَمًا وَرَثُوا الْعِلْمَ فَمَنْ أَخَذَهُ أَخَذَ بِحِظِّ وَافِرٍ

“Barangsiapa menempuh suatu jalan yang padanya dia mencari ilmu, maka Allah akan memudahkan dia menempuh jalan dari jalan-jalan (menuju) jannah, dan sesungguhnya para malaikat benar-benar akan meletakkan sayap-sayapnya untuk penuntut ilmu, dan sesungguhnya seorang penuntut ilmu akan dimintakan ampun untuknya oleh makhluk-makhluk Allah yang ada di langit dan yang ada di bumi, sampai ikan yang ada di tengah lautan pun memintakkan ampun untuknya. Dan sesungguhnya keutamaan seorang yang berilmu atas seorang yang ahli ibadah adalah seperti keutamaan bulan pada bulan purnama atas seluruh bintang, dan sesungguhnya ulama adalah pewaris para nabi dan para nabi tidaklah mewariskan dinar ataupun dirham, akan tetapi mereka hanya akan mewariskan ilmu, maka barangsiapa yang mengambilnya maka sesungguhnya dia telah mengambil bagian yang sangat banyak.”

Pada hadist yang diriwayatkan oleh Abu Dawud telah dijelaskan bahwa seseorang yang mencari ilmu akan diberikan kemuliaan yang sangat besar. Mencari informasi sama halnya dengan mencari ilmu, Allah memberikan banyak keistimewaan bagi orang-orang yang mau mencari informasi agar menambah ilmu pengetahuannya. Mencari sebuah informasi sama halnya dengan mencari dan menggali pengetahuan untuk menambah ilmu sebagaimana telah diperintahkan oleh Allah SWT untuk terus menambah pengetahuan.

Menurut kamus besar Bahasa Indonesia (Depdikbud, 1988) Ilmu memiliki dua pengertian, yaitu :

1. Ilmu diartikan sebagai suatu pengetahuan tentang suatu bidang yang disusun secara sistem menurut metode-metode tertentu, yang dapat digunakan untuk menerapkan gejala-gejala tertentu dibidang (pengetahuan) tersebut, seperti ilmu hukum, ilmu pendidikan, ilmu ekonomi dan sebagainya.

2. Ilmu diartikan sebagai pengetahuan atau kepandaian, tentang soal duniawi, akhirat, lahir, bathin, dan sebagainya, seperti ilmu akhirat, ilmu akhlak, ilmu bathin, ilmu sihir, dan sebagainya.

Pada hasil uji coba penelitian ini informasi yang didapatkan adalah tentang deskripsi, nama, suku, cara pemakaian, manfaat serta penelitian dari tanaman obat, informasi tersebut merupakan hasil dari *observasi* para peneliti tanaman obat yang dalam penelitiannya menggunakan metode-metode pengujian dalam bidang keilmuannya, Dalam hal ini informasi yang didapatkan dalam penelitian ini dapat di pandang sebagai ilmu pengetahuan.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Aplikasi *mobile* ensiklopedi tanaman obat indonesia berbasis berbasiskan *semantic search* dengan menggunakan Algoritma Boyer-Moore sebagai algoritma *string matching*. Aplikasi ini melakukan pencarian berdasarkan konteks kalimat. Berdasarkan hasil uji coba menggunakan 25 kalimat pencarian diketahui keakuratan pencarian pada aplikasi ini adalah 95%. Dari hasil uji coba dapat diketahui apabila semakin banyak tingkatan ontologi serta semakin banyak metadata yang digunakan tingkat keakuratannya akan semakin tinggi.

5.2 Saran

Proses pencarian dilakukan dengan menyaring kata yang berhubungan dengan tanaman obat dari kalimat yang dimasukkan oleh user, namun proses input kata yang digunakan untuk penyaringan masih dimasukkan secara manual, sehingga bisa dikembangkan menjadi *auto detect*. Pada saat proses pencarian SPO dari table ntriple pencarian masih mengacu pada objek saja sehingga proses pencarian belum sepenuhnya maksimal, untuk dalam hal ini dapat dilakukan pengembangan sehingga dapat mengkombinasikan pencarian dengan menggunakan subjek, predikat, dan objek.



DAFTAR PUSTAKA

- Anonim. 2012. *Tanaman*. Wikipedia Bahasa Indonesia: www.wikipedia.com. Diakses 31 Oktober 2012.
- Berners-Lee, T., Hendler, J., Lasilla, O. 2001. *The Semantic Web*, American Scientific.
- Brickley, D., Guha, R., & McBride, B. 2004. *RDF Vocabulary Description Language 1.0: RDF Schema*. W3C: <http://www.w3.org/TR/rdf-schema/>. Diakses 2 Agustus 2012.
- Chiquita, B Christabella. *Penerapan Algoritma Boyer Moore-Dynamic Programming untuk Layanan Auto-Complete dan Auto-Correct*. Makalah IF3051 Strategi Algoritma – Sem. I Tahun 2011/2012.
- Dalimartha, d. 1999. *Atlas Tumbuhan Obat Indonesia jilid 1,2,3,4*. Jakarta: Trubus Agriwidya.
- Edward G.R., *Algoritma Boyer-Moore*, <http://edwardgr.wordpress.com/2009/01/06/algoritma-boyer-moore/>. Diakses pada 25 Oktober 2012.
- Harian Kompas. 2013. *11 Makanan Pembunuh Kecerdasan*. Rosmha Widiyani : <http://health.kompas.com/read/2013/08/04/2120007/11.Makanan.Pembunuh.Kecerdasan>. Diakses 31 Oktober 2013.
- Jena, <http://jena.apache.org/>. Diunduh pada 03 April 2012
- K. Smith, M., Welty, C., & McGuinness, D. L. 2004. *OWL Web Ontology Language Guide*. W3C Recommendation: <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>. Diakses 25 April 2012.
- Poerwadarminta, W. 1993. *Kamus Umum Bahasa Indonesia*. Jakarta: Balai Pustaka.
- Protégé, <http://protege.stanford.edu/download/download.html>. Diunduh pada 25 Juli 2012.
- S., A. G., Singgih, M., & Dh., H. 2010. *Prospek Tumbuhan Indonesia dalam Kesehatan dan Permasalahannya*. Ikatan Apoteker Indonesia: <http://www.ikatanapotekerindonesia.net/articles/pharma-update/national-pharmacy/340-prospek-tumbuhan-indonesia-dalam-kesehatan-dan-permasalahannya.html>. Diakses 31 Oktober 2013.

- Seaborne, A., Margulies, B., Dollin, C., Steer, D., Reynolds, D., Dickinson, I., et al. 2011. *Jena architecture overview*. Apache Jena: http://jena.apache.org/about_jena/architecture.html. Diakses 25 April 2013.
- Savitri, E. S. 2008. *Rahasia Tumbuhan Berkhasiat Obat Perspektif Islam*. Malang: UIN Press.
- Siahaan, D., & Prapanca, A. 2009. International Journal of Information Technology. *RDFGraph: New Data Modeling Tool for Semantic Web* , 259.
- Sialllagan, I. 2009. *SemanticWeb: Caragooglememahamikemauan anda*, Chip Simple HTML DOM Parser, <http://simplehtmldom.sourceforge.net/>. Diunduh pada 23 Mei 2013.
- Smore, <http://www.mindswap.org/>. Diunduh pada 25 Juli 2012.
- Sphider. <http://www.sphider.eu/download.php>. Diunduh Pada 25 September 2012.
- Sriwijaya Post. 2012, *Di Indonesia, PonselPintarCumauntuk SMS-Media Sosial*. Team Sriwijaya Post: <http://palembang.tribunnews.com/2012/06/07/di-indonesia-ponsel-pintar-cuma-untuk-sms-media-sosial>. Diakses 31 Oktober 2013
- Yu, Liyang. 2007. *Introduction to the Semantic Web and Semantic Web Services*. Chaapman & Hall/CRC
- Wijayakusuma, P. H. 2000. *Ensiklopedia Milenium Tumbuhan Berkhasiat Obat Indonesia*. Jakarta: Prestasi Insan Indonesia.
- W3C, *Available online Accessed*: <http://www.w3.org/standards/semanticweb/>., Diakses 11 Januari 2012.
- W3techs, *Usage of content management systems for websites*. W3techs http://w3techs.com/technologies/overview/content_management/all. Diakses 22 September 2013