

LAPORAN PENELITIAN



AKSES NODE DOKUMEN XML MENGGUNAKAN XPATH UNTUK PEROLEHAN DATA PADA BASISDATA PERUNDANGAN PEMERINTAH KOTA SEMARANG

Oleh :

1. Imam Husni Al Amin, ST / YU.2.04.04.068 (Ketua)
2. Setyawan Wibisono, S.Kom, M.Cs / 197306072005011001 (Anggota)
3. Fatkhul Amin, ST / YU.2.02.10.044 (Anggota)
4. Feri Dwi Astuti / 08.01.53.0056 (Anggota)
5. Agustina Wijaya / 08.01.53.0067 (Anggota)

**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS STIKUBANK (UNISBANK) SEMARANG
AGUSTUS 2012**

**HALAMAN PENGESAHAN
LAPORAN PENELITIAN**

1. Judul Pelatihan : Akses Node Dokumen XML Menggunakan XPath untuk Perolehan Data pada Basisdata Perundangan Pemerintah Kota Semarang
2. Bidang Ilmu : Komputer (Teknik Informatika)
3. Ketua Peneliti
 - a. Nama Lengkap : Imam Husni Al Amin, ST
 - b. Jenis Kelamin : Laki-laki
 - c. NIP : YU.2.04.04.068
 - d. Disiplin Ilmu : Ilmu Komputer
 - e. Pangkat / Golongan : Penata / IIIC
 - f. Jabatan Fungsional : Lektor
 - g. Fakultas / Jurusan : Teknologi Informasi / Teknik Informatika
 - h. Alamat Kampus : Jl. Trilomba Juang No. 1 Semarang
 - i. Telp / Faks / Email : 024-8311668/024-8443240/info@unisbank.ac.id
 - j. Alamat Rumah : Jl. Gemah Selatan II No 56, Semarang
 - k. Telp / Faks / Email : 085641524028/ - / imam_husni@yahoo.co.id
4. Jumlah Anggota Peneliti : 4 orang
 - Nama Anggota I : Setyawan Wibisono, S.Kom, M.Cs
 - Nama Anggota II : Fatkhul Amin, ST
 - Mahasiswa Yang Terlibat : 1. Feri Dwi Astui
2. Agustina Wijaya
5. Lokasi Penelitian : Laboratorium Komputer Unisbank Semarang
6. Jangka waktu Penelitian : 4 (empat) bulan, 21 Desember 2011 - 21 April 2012
7. Jumlah biaya yang diusulkan : Rp. 3.000.000,00

Semarang, 3 Agustus 2012

Mengetahui,
Dekan Fakultas Teknologi Informasi

Ketua Peneliti,

(Dwi Agus Diartono, S.Kom., M.Kom)
NIY. Y.2.90.03.054

(Imam Husni Al Amin, ST)
NIY. YU.2.04.04.068

Menyetujui,
Ketua LPPM Unisbank

(Dr. Dra. Lie Liana, M.MSI)
NIY. Y.2.92.07.085

KATA PENGANTAR

Puji syukur kepada Allah SWT yang telah memperkenankan penelitian dengan judul : Akses Node Dokumen XML Menggunakan XPath untuk Perolehan Data pada Basisdata Perundangan Pemerintah Kota Semarang dapat selesai dengan baik. Kiranya penelitian ini dapat bermanfaat bagi sivitas akademika Universitas Stikubank Semarang, khususnya dosen, instruktur dan mahasiswa Universitas Stikubank Semarang. Dengan selesainya penelitian ini kami mengucapkan banyak terima kasih kepada:

1. Dekan Fakultas Teknologi Informasi, Universitas Stikubank yang telah memberikan kepercayaan dan kesempatan.
 2. Ketua Program Studi Teknik Informatika Universitas Stikubank yang telah memberikan dorongan, bantuan dan fasilitas.
 3. Semua staf pengajar Program Studi Teknik Informatika Universitas Stikubank yang telah memberikan sumbang sarannya demi terselesaikannya penelitian ini.
- Segala kritik dan saran sangat kami harapkan demi perbaikan penelitian ini di masa yang akan datang.

Semarang, Agustus 2012

Penyusun

INTISARI

AKSES NODE DOKUMEN XML MENGGUNAKAN XPATH UNTUK PEROLEHAN DATA PADA BASISDATA PERUNDANGAN PEMERINTAH KOTA SEMARANG

Imam Husni Al Amin, ST, Fatkhul Amin, ST, Setyawan Wibisono, S.Kom, M.Cs,
Feri Dwi Astuti, Agustina Wijaya

Fakultas Teknologi Informasi, Program Studi Teknik Informatika
Universitas Stikubank (UNISBANK) Semarang

Saat ini sebagian dokumen perundangan Pemerintah Kota Semarang masih banyak dalam bentuk hardcopy, belum dapat secara mudah diakses masyarakat melalui teknologi internet. Produk perundangan Pemerintah Kota Semarang yang dimaksud di sini adalah Peraturan Daerah Kota Semarang, Peraturan Walikota dan Surat Edaran Walikota

Dalam penelitian ini ada beberapa langkah yang dilakukan, yaitu: merancang XPath dengan memperhatikan node yang akan diambil. Xpath ini akan dapat diimplementasikan dalam XQuery untuk memperoleh data pada dokumen dalam format XML dari suatu koleksi dokumen perundangan Pemerintah Kota Semarang.

Program aplikasi untuk perolehan data pada basisdata perundangan Pemerintah Kota Semarang adalah memanfaatkan akses node pada sebuah XPath dokumen XML dalam membuat basisdata XML serta menggunakan XPath untuk memperoleh informasi sesuai query yang dimasukkan. Query yang dimasukkan adalah suatu kalimat dalam bahasa Indonesia. Sistem akan membuat sebuah jalur (path) yang disebut Xpath untuk mengakses node yang dituju sesuai dengan kalimat yang diinputkan.

Cara memperoleh data menggunakan cara menginputkan suatu kalimat dalam bahasa Indonesia, dengan tata bahasa yang telah ditentukan. Pada aplikasi ini juga diberikan contoh cara memberikan kalimat input, maupun pernyataan perintah untuk memperoleh data. Output dari aplikasi ini dalam bentuk tabel kepada user, jika input kalimat yang diberikan sudah benar dan data tersedia sesuai permintaan. Data yang ditampilkan adalah node dari hasil pengambilan data berdasarkan XPath. Jika format kalimat tidak sesuai dengan aturan maka yang dimunculkan oleh aplikasi ini adalah sebuah peringatan kesalahan.

Kata kunci : node, perundangan, XML, XQuery, XPath,

ABSTRACTION

XQUERY IMPLEMENTATION IN THE XML DATABASE AT LEGISLATION INFORMATION SYSTEM PROVINCIAL GOVERNMENT OF CENTRAL JAVA

Setyawan Wibisono, S.Kom, M.Cs, Imam Husni Al Amin, ST
Sugiyanto, S.Kom, Susi Susanti, Muhamad Irfan Thobib

Faculty of Information Technology, Informatics Engineering Program
Stikubank University (UNISBANK) Semarang

Today most government documents and regulations are still a lot of Semarang in hardcopy, can not be easily accessible to the public via internet technology. Product Semarang City Government legislation in question here is the local regulation of Semarang, Regulations and Circular Letter of the Mayor Mayor In this study there are several steps taken, namely: designing XPath with respect to the node that will be taken. This xpath will be implemented in XQuery to obtain data on the document in XML format of a document collection Semarang City Government legislation.

Program application for the acquisition of data on the database law Semarang Municipality is utilizing the access node in an XML document in XPath make XML databases and use XPath queries to obtain the appropriate information is entered. Entered query is a sentence in the Indonesian language. The system will create a path (path) is called XPath to access the destination node is entered in accordance with the sentence.

How to obtain the data using menginputkan a sentence in the Indonesian language, with grammar that has been determined. In this application also provided examples of how to provide input sentence, or statement of a command to obtain data. The output of this application in tabular form to the user, if the input sentence given is correct and the data are available on request. Data shown are the nodes of the retrieval of data based on XPath If the format does not comply with the rules of the sentence is raised by this application is an error warning.

Keywords : legislation, node, XML, XQuery, XPath,

DAFTAR ISI

Halaman Judul	i
Halaman Pengesahan	ii
Kata Pengantar	iii
Intisari	iv
Abstract	v
Daftar Isi	vi
Daftar Gambar	vii
BAB I PENDAHULUAN	1
BAB II TUJUAN DAN MANFAAT	4
BAB III TELAAH PUSTAKA	5
3.1. XML	5
3.2. XML Schema	6
3.3. XPath.....	7
3.4. XML Query Languages	9
3.5. Database XML	10
3.6. Pembentukan Perundangan Daerah	12
BAB IV METODOLOGI PENELITIAN	14
BAB V HASIL DAN PEMBAHASAN	17
5.1. Deskripsi Sistem	14
5.2. Bagian – Bagian Sistem	19
5.2.1. Scanner	19
5.2.2. Parser	19
5.2.3. Translator	20
5.2.4. Evaluator	20
5.3. Struktur Data	21
5.4. Identifikasi Tipe – Tipe Query	23
5.5. Implementasi Basisdata	25
5.6. Implementasi Komponen Pemroses Bahasa	27
5.6.1. DisplayQueryNlp.java	27
5.6.2. Scanner	28
5.6.3. Parser	28
5.6.4. Translator	28
5.6.5. Evaluator	29
5.6.6. DemoFrame.java	30
5.6. Tampilan Aplikasi	31
BAB VI KESIMPULAN DAN SARAN	34
6.1. Kesimpulan	34
6.2. Saran	34
DAFTAR PUSTAKA	35

DAFTAR GAMBAR

Gambar 5.1.	Diagram Aplikasi Query Processing Basisdata XML	18
Gambar 5.2.	XML Schema	22
Gambar 5.3.	XML Grid	22
Gambar 5.4.	Struktur data Perundangan Pemerintah Kota Semarang dalam format XML	26
Gambar 5.5.	Tampilan data perundangan dalam format XML.....	26
Gambar 5.6.	Koneksi database	27
Gambar 5.7.	Pesan kesalahan	27
Gambar 5.8.	Penggunaan Utility Scanner.....	28
Gambar 5.9.	Deklarasi Parser	28
Gambar 5.10.	Translator	28
Gambar 5.11.	Evaluator	30
Gambar 5.12.	DemoFrame.....	30

BAB I

PENDAHULUAN

Mesin pencari di internet dalam dekade ini sudah menjadi bagian terpenting untuk memperoleh data maupun informasi. Perolehan informasi hukum dan perundangan telah banyak melibatkan mesin pencari. Dokumen hukum dan perundangan secara global tidak terlalu sulit untuk dicari melalui internet. Masalah yang sering muncul adalah kelemahan dalam memelihara arsip hukum dan perundangan menyebabkan data dan informasi tentang perundangan menjadi tidak didapatkan. Arsip-arsip perundangan yang muncul ketika dilakukan pencarian melalui internet adalah arsip-arsip yang relatif baru di awal tahun 1990-an, dimana mulai digunakannya komputer sebagai alat bantu pengarsipan.

Sampai saat ini informasi tentang perundangan masih relatif sulit untuk ditemukan, padahal perundangan merupakan suatu dasar bagi pemerintah dalam menjalankan roda pemerintahan serta masyarakat dalam menjalankan usaha dan urusan sehari-hari. Memang secara formal informasi tentang perundangan haruslah dapat diakses oleh masyarakat umum, tetapi dengan kenyataan yang sekarang terjadi, maka hal tersebut masihlah jauh dari kenyataan. Sebagai gambaran bahwa arsip-arsip perundangan pada pemerintahan di daerah masih relatif sulit diperoleh, karena arsip-arsip dapat berada di unit kerja yang berlainan, bukan pada satu unit yang mengurus arsip perundangan. Walaupun pemerintah di daerah sudah mulai terbuka dengan mempublikasi arsip perundangan di daerah dengan membentuk jaringan dokumen dan informasi pada biro hukum di daerah. Maka diperlukan suatu upaya untuk menyimpan arsip-arsip perundangan secara lebih tertib dan terdokumentasi secara elektronik sehingga mudah dalam melakukan publikasi.

Dalam penyimpanan arsip perundangan secara elektronik maka diperlukan penyimpanan dalam basisdata yang aplikatif. Tidaklah sulit untuk memilih database engine yang aplikatif dan reliabel saat ini. Mulai dari database engine yang berlisensi sampai dengan database engine yang tidak berbayar. Begitu juga untuk perolehan data yang tersimpan, dalam teknologi basisdata sekarang ini SQL

(*Structured Query Language*) sudah jamak digunakan. SQL adalah suatu cara untuk memperoleh data dari suatu basis data dengan menggunakan bahasa standar. Untuk dapat memperoleh data dari suatu basisdata, maka pengguna harus menuliskan perintah SQL dalam aturan yang telah ditentukan.

Saat ini sebagian dokumen perundangan Pemerintah Kota Semarang masih banyak dalam bentuk hardcopy, belum dapat secara mudah diakses masyarakat melalui teknologi internet. Produk perundangan Pemerintah Kota Semarang yang dimaksud di sini adalah Peraturan Daerah Kota Semarang, Peraturan Walikota dan Surat Edaran Walikota

Beberapa produk perundangan Pemerintah Kota Semarang didokumentasikan dalam bahasa Indonesia tanpa memiliki format maupun struktur yang standar. Sehingga ketika dibutuhkan suatu penyimpanan produk perundangan tersebut dalam suatu sistem, maka harus dibuat suatu struktur data tentang produk perundangan tersebut. Sehingga pembuatan struktur dokumen produk perundangan Pemerintah Provinsi Jawa Tengah yang standar menjadi hal yang utama, sebelum proses analisis terhadap dokumen tersebut dilakukan. Format XML menjadi format yang cocok sebagai struktur standar dokumen perundangan Pemerintah Provinsi Jawa Tengah karena XML merupakan standar penulisan dokumen yang dapat memisahkan struktur dan isi dari dokumen (Mawadah dan Budi, 2007).

Dalam hal ini yang diperlukan adalah suatu sistem yang cerdas (*Intelligent System*) yang secara otomatis sistem ini dapat :

1. Menyimpan dokumen perundangan Kota Semarang yang telah disahkan ke dalam basisdata dengan format XML.
2. Menampilkan data dari suatu pencarian spesifik tentang dokumen perundangan Pemerintah Kota Semarang.

Dari permasalahan yang muncul inilah maka dikembangkan suatu metode untuk menangani masalah-masalah di atas, dalam hal ini contoh kasus yang diangkat adalah perolehan informasi dengan domain produk perundangan Pemerintah Kota Semarang dengan tahap :

1. Penyiapan dokumen

Pada tahap ini akan dilakukan manipulasi teks pada dokumen yang selanjutnya tiap dokumen akan direpresentasikan dalam format XML.

2. Pengolahan data dalam format XML

Untuk memperoleh informasi dengan pencarian diperlukan sistem pengolahan informasi dalam format data XML.

Dalam tahap penyiapan dan pengolahan data diperlukan suatu bahasa Query untuk merepresentasikan data. Dalam hal ini basisdata yang digunakan adalah basisdata dalam format XML, sehingga bahasa Query yang digunakan adalah XQuery. Sedangkan untuk mengakses data (node) dalam data XML digunakan XPath.

Berdasarkan latar belakang di atas, maka permasalahan yang dapat dirumuskan adalah :

1. Bagaimana merancang dokumen dalam format XML dari suatu koleksi dokumen perundangan Pemerintah Kota Semarang.
2. Bagaimana merancang XPath yang dapat diimplementasikan dalam XQuery, sehingga dapat digunakan untuk perolehan informasi dokumen perundangan Pemerintah Kota Semarang.

Dalam penelitian ini ada beberapa pembatasan masalah yang dilakukan, yaitu: merancang XPath yang dapat diimplementasikan dalam XQuery pada dokumen dalam format XML dari suatu koleksi dokumen perundangan Pemerintah Kota Semarang.

BAB II

TUJUAN DAN MANFAAT

Tujuan yang ingin dicapai dalam penelitian ini adalah :

1. Mengimplementasikan XQuery dalam merancang dan membuat dokumen perundangan Pemerintah Kota Semarang.
2. Mengimplementasikan XQuery dalam membuat sistem perolehan informasi dokumen perundangan Pemerintah Kota Semarang .

Manfaat yang diharapkan dari penelitian ini adalah memudahkan pencarian informasi dengan domain dokumen perundangan Pemerintah Kota Semarang dalam format XML. Hasil penelitian bisa digunakan untuk melakukan penyimpanan dokumen dengan format text dan dapat digunakan untuk pencarian dokumen relevan berdasarkan kata kunci tertentu yang diinputkan pengguna.

Hasil penelitian ini diharapkan dapat digunakan sebagai acuan oleh praktisi, mahasiswa, staf administrasi bagian hukum Pemerintah Kota Semarang dalam pencarian informasi peraturan perundangan pada tingkat Pemerintah Kota Semarang. Hal ini bermanfaat agar para praktisi, mahasiswa, staf administrasi bagian hukum Pemerintah Kota Semarang lebih mudah dalam pencarian informasi dengan domain dokumen perundangan Pemerintah Kota Semarang. Hasil penelitian bisa digunakan untuk melakukan penyimpanan dokumen dengan format XML dan dapat digunakan untuk pencarian dokumen relevan berdasarkan kata kunci tertentu yang diinputkan pengguna..

BAB III

TELAAH PUSTAKA

3.1. XML

Extensible Markup Language (XML) adalah format berbasis teks sederhana untuk mewakili informasi terstruktur: dokumen, data, konfigurasi, buku, transaksi, faktur, dan banyak lagi. Format ini berasal dari format standar yang lebih ada sebelumnya yang disebut SGML (ISO 8879), agar lebih sesuai digunakan untuk Web.

XML adalah salah satu format yang paling banyak digunakan untuk berbagi informasi yang terstruktur hari ini: antar program, antar manusia, antara komputer dan manusia, baik lokal maupun melalui jaringan. Berikut ini adalah sebuah contoh singkat:

```
<part number="1976">
  <name>Windscreen Wiper</name>
  <description>The Windscreen wiper
    automatically removes rain
    from your windscreen, if it
    should happen to splash there.
    It has a rubber <ref part="1977">blade</ref>
    which can be ordered separately
    if you need to replace it.
  </description>
</part>
(http://www.w3.org)
```

XML memiliki sejumlah keunggulan dibandingkan format lainnya. Beberapa keuntungan dari XML adalah:

1. Redundansi

Markup XML sangat verbose. Misalnya, setiap tag penutup harus diberikan, seperti `</description>` dalam contoh. Ini memungkinkan komputer menangkap kesalahan umum seperti format bersarang yang salah.

2. Self-describing

Pembacaan XML (merupakan format berbasis teks) dan adanya nama elemen dan atribut dalam XML berarti bahwa orang melihat dokumen XML sering bisa

mendapatkan kepala mulai pada pemahaman format (dan juga membantu orang untuk menemukan kesalahan)

3. Efek Jaringan dan Janji XML

Setiap dokumen XML dapat dibaca dan diproses oleh alat XML apapun. Tentu saja, beberapa alat XML mungkin ingin spesifik markup XML, tetapi format XML itu sendiri dapat dibaca oleh parser XML: Anda tidak bisa mengatakan, ini dokumen XML hanya untuk diproses oleh ini-dan-alat tersebut.

Ini berarti bahwa setiap dokumen XML baru meningkatkan nilai dari setiap dokumen XML lain, dan setiap alat XML, dan setiap alat XML baru meningkatkan nilai dari setiap dokumen XML dan karenanya setiap alat lainnya. Hari ini, XML merupakan format yang paling banyak digunakan dari jenisnya di mana saja di dunia. (<http://www.w3.org>)

3.2. XML Schema

XML Schema adalah bahasa untuk mengekspresikan kendala tentang dokumen XML. Ada beberapa bahasa skema yang berbeda digunakan secara luas, tapi yang utama adalah Document Type Definition (DTD), Relax-NG, Schematron dan W3C XSD (XML Schema Definition).

Schema dapat digunakan :

1. Untuk memberikan daftar dari elemen dan atribut dalam kosakata.
2. Untuk mengasosiasikan jenis, seperti integer, string, dll, atau lebih khusus seperti `hatsize`, `sock_colour`, dll, dengan nilai yang ditemukan dalam dokumen.
3. Untuk membatasi mana elemen dan atribut dapat muncul, dan apa yang dapat muncul dalam elemen-elemen, seperti mengatakan bahwa judul bab terjadi di dalam sebuah bab, dan bab harus terdiri dari judul bab diikuti dengan satu atau lebih paragraf teks.
4. Untuk menyediakan dokumentasi yang baik dibaca manusia dan mesin-processable.
5. Untuk memberikan deskripsi formal dari satu atau lebih dokumen.

Informasi dalam dokumen skema sering digunakan oleh XML-aware sistem editing sehingga mereka dapat menawarkan pengguna elemen yang paling

mungkin terjadi pada setiap lokasi yang diberikan dalam dokumen. Memeriksa dokumen terhadap Skema ini dikenal sebagai memvalidasi terhadap skema itu, untuk DTD ini hanya memvalidasi, tapi untuk jenis lainnya jenis skema itu disebutkan, seperti Validasi XSD atau validasi Relax-NG.

Memvalidasi terhadap skema merupakan komponen penting dari jaminan kualitas. Service Modeling Language (SML) memberikan kerangka untuk menghubungkan dokumen XSD ganda untuk satu atau lebih dokumen dalam episode validasi tunggal. Sejak XSD mendukung relasi dengan tipe data elemen dan isi atribut, juga digunakan untuk data yang mengikat, yaitu untuk komponen perangkat lunak yang membaca dan menulis representasi XML pemrograman komputer berbahasa objek. (<http://www.w3.org>)

3.3. XPath

XPath adalah bahasa non-XML digunakan untuk mengidentifikasi bagian-bagian tertentu dari dokumen XML. XPath menunjukkan node oleh posisi, posisi relatif, jenis, isi, dan kriteria lainnya. XSLT menggunakan ekspresi XPath untuk mencocokkan dan memilih elemen tertentu dalam dokumen masukan untuk menyalin ke dalam dokumen output atau diproses lebih lanjut. XPointer menggunakan ekspresi XPath untuk mengidentifikasi titik tertentu dalam atau bagian dari dokumen XML yang merupakan link ke XLink.

Ekspresi XPath juga dapat mewakili angka, string, atau Booleans, sehingga XSLT stylesheet melakukan aritmatika sederhana untuk penomoran dan referensi silang angka, tabel, dan persamaan. String manipulasi dalam XPath memungkinkan XSLT melakukan tugas-tugas seperti membuat judul sebuah bab dalam judul huruf besar, tapi kasus campuran dalam referensi dalam teks tubuh.

XPath menggunakan jalur ekspresi untuk memilih node atau node-set dalam dokumen XML. Node dipilih dengan mengikuti jalur atau langkah. XPath adalah sintaks untuk mendefinisikan bagian dari dokumen XML. Ekspresi XPath paling berguna adalah jalur lokasi. Sebuah jalur lokasi menggunakan setidaknya satu langkah lokasi untuk mengidentifikasi satu set node dalam dokumen. Set ini mungkin kosong, berisi node tunggal, atau mengandung beberapa node. Node ini

bisa menjadi elemen, atribut, namespace, teks, komentar, pemrosesan instruksi, node root, atau kombinasi dari mereka. (Harold dan Means, 2001)

Path lokasi yang paling sederhana adalah salah satu yang memilih simpul akar dokumen. Jalan ini hanya garis miring /. Akan terlihat bahwa banyak sintaks XPath memang sengaja dipilih untuk menjadi mirip dengan sintaks yang digunakan oleh shell Unix. Simbol / adalah akar dari filesystem Unix dan / adalah node root dari suatu dokumen XML. Misalnya, template XSLT menggunakan pola XPath / untuk mencocokkan pohon masukan seluruh dokumen dan membungkusnya dalam sebuah elemen html:

```
<xsl:template match="/">
  <html><xsl:apply-templates/></html>
</xsl:template>
```

Garis miring / adalah path lokasi absolut karena tidak peduli apa node konteks adalah, di mana pun Anda berada di dokumen masukan ketika template ini diterapkan, selalu berarti hal yang sama: node root dokumen. Ini adalah relatif terhadap dokumen Anda memproses, tetapi tidak untuk apa-apa dalam dokumen itu. (Harold dan Means, 2001).

Sebagai contoh untuk dokumen XML di bawah ini.

```
<ujian>
  <universitas>UNIVERSITAS STIKUBANK</universitas>
  <fakultas>FAKULTAS TEKNOLOGI INFORMASI</fakultas>
  <kode>A.2.5.1006</kode>
  <matakuliah>SISTEM OPERASI</matakuliah>
  <progdi>SISTEM INFORMASI</progdi>
  <jenjang>S1</jenjang>
  <periode>AKHIR SEMESTER</periode>
  <semester>GASAL</semester>
  <tahunakademik>2007/2008</tahunakademik>
  <kelompok>PAGI</kelompok>
  <hari>JUMAT</hari>
  <tanggal>18 JANUARI 2008</tanggal>
  <jam>08:00</jam>
  <>waktu>90 MENIT</waktu>
  <sifat>TUTUP BUKU</sifat>
  <perintah>KERJAKAN SEMUA SOAL DI BAWAH INI</perintah>
  <deskripsi>
    <no>1</no>
    <soal>APA YANG MENJADI TUJUAN KEAMANAN SISTEM DAN
      JELASKAN JENIS-JENIS GANGGUAN PADA KEAMANAN
      SISTEM KOMPUTER!
    </soal>
  <bobot>20%</bobot>
```

```
</deskripsi>  
</ujian>
```

3. 4. XML Query Languages

Tidak seperti template berbasis bahasa query dan bahasa query berbasis SQL, yang hanya dapat digunakan dengan database relasional, bahasa query XML dapat digunakan atas setiap dokumen XML. Untuk menggunakan dengan database relasional, data dalam database harus dimodelkan sebagai XML, sehingga memungkinkan permintaan atas dokumen XML virtual.

Dengan XQuery, baik pemetaan berbasis tabel atau object-relational mapping dapat digunakan. Jika pemetaan berbasis tabel yang digunakan, setiap tabel diperlakukan sebagai dokumen terpisah dan bergabung antara tabel (dokumen) yang ditentukan dalam permintaan sendiri, seperti di SQL. Jika object-relational mapping digunakan, hirarki tabel diperlakukan sebagai satu dokumen dan bergabung ditentukan dalam pemetaan. Tampaknya mungkin bahwa berbasis tabel pemetaan akan digunakan dalam implementasi yang paling atas database relasional, karena ini tampaknya mudah untuk diterapkan dan lebih akrab bagi pengguna SQL.

Dengan XPath, sebuah object-relational mapping harus digunakan untuk melakukan query di lebih dari satu tabel. Hal ini karena XPath tidak mendukung bergabung di seluruh dokumen. Jadi, jika pemetaan berbasis tabel yang digunakan, akan ada kemungkinan untuk query hanya satu meja pada suatu waktu.

Hal ini juga memungkinkan untuk menyimpan data dalam dokumen XML dalam database XML asli. Ada beberapa alasan untuk melakukan ini. Yang pertama adalah ketika data Anda semi-terstruktur. Artinya, ia memiliki struktur biasa, tapi struktur yang bervariasi cukup bahwa pemetaan itu ke hasil database relasional baik dalam sejumlah besar kolom dengan nilai null (yang limbah spasi) atau sejumlah besar tabel (yang tidak efisien). Meskipun data semi-terstruktur dapat disimpan dalam database berorientasi objek dan hirarki, Anda juga dapat memilih untuk menyimpannya dalam database XML asli dalam bentuk dokumen XML.

Alasan kedua untuk menyimpan data dalam database XML asli adalah kecepatan pengambilan. Tergantung pada bagaimana database XML asli secara fisik menyimpan data, mungkin bisa mengambil data jauh lebih cepat dari database relasional. Alasan untuk ini adalah bahwa strategi penyimpanan yang digunakan oleh beberapa database XML asli menyimpan dokumen bersama seluruh fisik atau menggunakan fisik (bukan logis) pointer antara bagian-bagian dari dokumen. Hal ini memungkinkan dokumen yang akan diambil baik tanpa bergabung atau dengan fisik bergabung, yang keduanya lebih cepat dari logis bergabung digunakan oleh database relasional.

Sebagai contoh, perhatikan dokumen order penjualan di atas. Dalam database relasional, ini akan disimpan dalam empat meja - SalesOrders, Produk, Pelanggan, dan Bagian - dan mengambil dokumen akan memerlukan bergabung di tabel ini. Dalam sebuah database XML asli, seluruh dokumen akan disimpan ke dalam satu tempat pada disk, jadi mengambilnya atau fragmen dari itu memerlukan indeks pencarian tunggal dan tunggal membaca untuk mengambil data. Sebuah database relasional memerlukan empat indeks pencarian dan setidaknya empat membaca untuk mengambil data.

Kelemahan yang jelas dalam hal ini adalah bahwa kecepatan meningkat berlaku hanya ketika mengambil data dalam urutan yang akan disimpan pada disk. Jika Anda ingin mengambil pandangan yang berbeda dari data, seperti daftar pelanggan dan pesanan penjualan yang berlaku untuk mereka, kinerja mungkin akan lebih buruk daripada dalam database relasional. Dengan demikian, penyimpanan data dalam database XML asli untuk alasan kinerja hanya berlaku ketika satu tampilan data menonjol dalam aplikasi Anda.

(<http://www.rpbouret.com/xml/XMLAndDatabases.htm/#xmlquery>)

3.5. Database XML

Sebuah dokumen XML adalah database hanya dalam arti ketat dari istilah tersebut. Artinya, itu adalah kumpulan data. Dalam banyak hal, ini membuatnya tidak berbeda dari file lainnya, semua file berisi data dari beberapa macam. Sebagai format "database", XML memiliki beberapa keuntungan. Sebagai contoh, self-

describing (markup menjelaskan struktur dan nama jenis data, meskipun tidak semantik), portabel (Unicode), dan dapat menggambarkan data dalam struktur pohon atau grafik. Ia juga memiliki beberapa kelemahan. Sebagai contoh, adalah verbose dan akses ke data yang lambat karena parsing dan konversi teks.

Teknologi XML dan sekitarnya merupakan "database" dalam pengertian yang lebih luas yaitu, sebuah sistem manajemen database (DBMS). Di sisi positifnya, XML menyediakan banyak hal yang ditemukan dalam database: penyimpanan (XML dokumen), skema (DTD, XML Schemas, RELAX NG, dan sebagainya), query bahasa (XQuery, XPath, XQL, XML-QL, Quilt , dll), pemrograman antarmuka (SAX, DOM, JDOM), dan sebagainya. Kekurangannya adalah tidak memiliki banyak hal yang ditemukan di database nyata: penyimpanan yang efisien, indeks, keamanan, transaksi dan integritas data, akses multi-user, pemicu, permintaan di beberapa dokumen, dan sebagainya.

Jadi, meskipun dimungkinkan untuk menggunakan dokumen XML atau dokumen sebagai database dalam lingkungan dengan data dalam jumlah kecil, beberapa pengguna, dan persyaratan kinerja sederhana, ini akan gagal dalam lingkungan produksi yang paling, yang memiliki banyak pengguna, persyaratan data ketat integritas, dan kebutuhan untuk kinerja yang baik.

Adalah jauh lebih mudah untuk menciptakan sebuah bahasa XML kecil dan menulis aplikasi SAX untuk menafsirkan bahwa bahasa daripada untuk menulis parser untuk koma-delimited file. Selain itu, XML memungkinkan Anda untuk memiliki entri bersarang, sesuatu yang sulit untuk dilakukan dalam koma-delimited file. Namun, hal ini hampir tidak database, karena dibaca dan ditulis secara linear, dan kemudian hanya ketika aplikasi dijalankan dan berakhir. Contoh beberapa titik data yang canggih untuk dokumen XML mungkin cocok sebagai database adalah daftar kontak pribadi (nama, nomor telepon, alamat, dll), penanda browser, dan deskripsi dari MP3 Anda sudah dicuri dengan bantuan Napster . Namun, mengingat harga rendah dan kemudahan penggunaan database seperti dBASE dan Akses, tampaknya ada sedikit alasan untuk menggunakan dokumen XML sebagai database bahkan dalam kasus ini. Satu-satunya keuntungan

nyata dari XML adalah bahwa data portabel, dan ini kurang dari keuntungan dari tampaknya karena ketersediaan luas alat untuk serialisasi database sebagai XML.

(<http://www.rpbouret.com/xml/XMLAndDatabases.htm/#isxmladatabase>)

3.6 Pembentukan Perundangan Daerah

Perda sebagai jenis PUU nasional memiliki landasan konstitusional dan landasan yuridis dengan diaturnya kedudukan Perda dalam UUD 1945 Pasal 18 ayat (6), UU No.10 Tahun 2004 tentang Pembentukan Peraturan Perundang-undangan, UU No.32 Tahun 2004 tentang Pemerintahan Daerah termasuk perundang-undangan tentang daerah otonomi khusus dan daerah istimewa sebagai *lex specialis* dari UU No.32/2004[2]. Selain itu terkait dengan pelaksanaan wewenang dan tugas DPRD dalam membentuk Perda adalah UU No.27 Tahun 2009 tentang MPR, DPR, DPD, dan DPRD dan Peraturan Pemerintah No.16 Tahun 2010 tentang Pedoman Penyusunan Peraturan DPRD tentang Tata Tertib DPRD. Penting pula untuk diperhatikan UU No.28 Tahun 2009 tentang Pajak dan Retribusi sebagai pengganti UU No.18 Tahun 1997 sebagaimana telah diubah terakhir dengan UU No.34 Tahun 2004, dalam rangka pengendalian Perda tentang Pajak dan Retribusi dan UU No.26 Tahun 2007 tentang Penataan Ruang dalam rangka keterpaduan RTRW nasional, provinsi, dan kabupaten/kota.

Pasal 18 ayat (6) UUD 1945 menyatakan bahwa pemerintah daerah berhak menetapkan peraturan daerah dan peraturan-peraturan lain untuk melaksanakan otonomi daerah dan tugas pembantuan. Dalam kaitan ini maka sistem hukum nasional memberikan kewenangan atributif kepada daerah untuk menetapkan Perda dan peraturan daerah lainnya, dan Perda diharapkan dapat mendukung secara sinergis program-program Pemerintah di daerah.

Perda sebagaimana PUU lainnya memiliki fungsi untuk mewujudkan kepastian hukum (*rechtszekerheid*, *legal certainty*). Untuk berfungsinya kepastian hukum PUU harus memenuhi syarat-syarat tertentu antara lain konsisten dalam perumusan dimana dalam PUU yang sama harus terpelihara hubungan sistematis antara kaidah-kaidahnya, kebakuan susunan dan bahasa, dan adanya hubungan harmonisasi antara berbagai peraturan perundang-undangan.

Pengharmonisasian PUU memiliki urgensi dalam kaitan dengan asas peraturan perundang-undangan yang lebih rendah tidak boleh bertentangan dengan peraturan perundang-undangan yang lebih tinggi, sehingga hal yang mendasar dalam penyusunan rancangan peraturan daerah adalah kesesuaian dan kesinkronannya dengan PUU lainnya. (Murti, 2010)

BAB IV

METODOLOGI PENELITIAN

Metodologi yang digunakan pada penelitian ini adalah:

1. Obyek Penelitian

Obyek penelitian dari penelitian ini adalah dokumen perundangan Pemerintah Kota Semarang.

2. Data Yang diperlukan

Merupakan data yang mendukung dalam penelitian ini meliputi data primer dan data sekunder.

a. Data primer

Dokumen perundangan dalam bentuk hardcopy dan softcopy, berupa Perda Kota Semarang, Peraturan Walikota dan Surat Edaran Walikota Semarang diperoleh langsung dari Biro Hukum Pemerintah Kota Semarang.

b. Data Sekunder

Data yang diperoleh dengan membaca dan mempelajari referensi mengenai JSP (Java Server Pages), basisdata, format data XML dan konsep perancangan XQuery.

3. Teknik Pengumpulan Data

Pengumpulan data mempunyai tujuan mendapatkan data – data yang mempunyai keterkaitan dengan topik penelitian. Pengumpulan data dimaksudkan agar mendapatkan bahan – bahan sebagai dasar dalam perancangan dan pembuatan data dokumen Perundangan Pemerintah Kota Semarang dalam format XML. Maka teknik pengumpulan data yang dilakukan pada penelitian ini adalah sebagai berikut :

b. Perancangan data

Dengan melakukan pengamatan, pencatatan, perancangan dan pembuatan basisdata dokumen perundangan Pemerintah Kota Semarang dengan format data XML.

c. Studi pustaka

Teknik pengumpulan data dilakukan dengan studi pustaka yang berhubungan dengan analisis data, pemodelan sistem dan perancangan sistem aplikasi, dengan pengumpulan data dari bahan – bahan referensi, arsip, dan dokumen yang berhubungan dengan permasalahan dalam penelitian ini.

4. Metode Pengembangan

Penelitian ini menggunakan model *prototyping*. Di dalam model ini sistem dirancang dan dibangun secara bertahap dan untuk setiap tahap pengembangan dilakukan percobaan-percobaan untuk melihat apakah sistem sudah bekerja sesuai dengan yang diinginkan. Sistematika model *prototyping* terdapat pada Gambar 2 memperlihatkan tahapan pada *prototyping*.

Berikut adalah tahapan yang dilakukan pada penelitian ini dengan metode pengembangan *prototyping*

a. Analisa

Pada tahap ini dilakukan analisa tentang masalah penelitian dan menentukan pemecahan masalah yang tepat untuk menyelesaikannya.

b. Desain

Pada tahap ini dibangun rancangan sistem dengan beberapa diagram bantu seperti Data Flow Diagram.

c. Prototype

Pada tahap ini dibangun aplikasi berbasis web yang sesuai dengan disain dan kebutuhan sistem.

d. Pengujian

Pada tahap ini dilakukan pengujian pada pustaka fungsi yang sudah dibangun.

e. Evaluasi

Pada tahap ini dilakukan evaluasi apakah performa aplikasi sudah sesuai dengan yang diharapkan, apabila belum maka dilakukan penyesuaian-penyesuaian secukupnya.

f. Penyesuaian

Tahap ini dilakukan apabila pada evaluasi performa aplikasi kurang memadai dan dibutuhkan perbaikan, tahap ini melakukan penyesuaian dan perbaikan pada aplikasi sesuai dengan kebutuhan.

BAB V

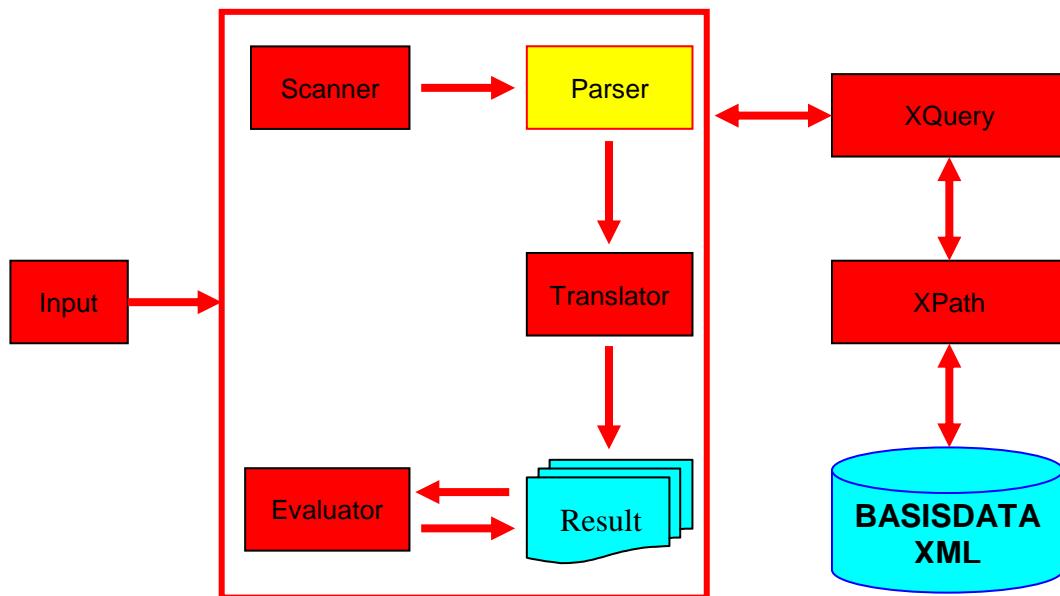
HASIL DAN PEMBAHASAN

5.1. Deskripsi sistem

Program aplikasi untuk perolehan data pada basisdata perundangan Pemerintah Kota Semarang adalah memanfaatkan akses node pada sebuah XPath dokumen XML dalam membuat basisdata XML serta menggunakan XPath untuk memperoleh informasi sesuai query yang dimasukkan. Query yang dimasukkan adalah suatu kalimat dalam bahasa Indonesia. Sistem akan membuat sebuah jalur (path) yang disebut Xpath untuk mengakses node yang dituju sesuai dengan kalimat yang diinputkan. Aplikasi ini dapat digunakan oleh praktisi hukum, pimpinan bagian hukum, staf bagian hukum dan staf administrasi lainnya pada Pemerintah Kota Semarang. akademik untuk membantu pekerjaan pencarian data dalam format XML yang berkaitan dengan data perundangan Pemerintah Kota Semarang. Aplikasi ini dapat dipergunakan sebagai alat bantu dalam pencarian data dengan domain perundangan pada Pemerintah Kota Semarang.

Dalam proses pencarian data pada sebuah basisdata, metoda yang lazim digunakan adalah menggunakan bahasa SQL untuk memperoleh data pada suatu tabel yang terdapat dalam suatu basisdata. Cara seperti ini adalah suatu cara yang banyak dilakukan oleh administrator dan operator suatu sistem yang melibatkan basisdata pada suatu sistem. Tetapi perolehan informasi dengan cara seperti ini hanya dipahami oleh orang yang mempunyai pengetahuan dan keterampilan dalam mengakses basisdata menggunakan perintah SQL.

Pada aplikasi ini cara memperoleh data menggunakan cara menginputkan suatu kalimat dalam bahasa Indonesia, dengan tata bahasa yang telah ditentukan. Pada aplikasi ini juga diberikan contoh cara memberikan kalimat input, maupun pernyataan perintah untuk memperoleh data.



Gambar 5.1. Diagram Aplikasi Query Processing Basisdata XML

Input yang dimasukkan adalah kalimat tanya atau kalimat perintah dalam bahasa Indonesia dengan tata bahasa yang telah ditentukan. Dalam aplikasi ini diberikan beberapa contoh kalimat yang sesuai dengan format aturan produksi.

Output dari aplikasi ini dalam bentuk tabel kepada user, jika input kalimat yang diberikan sudah benar dan data tersedia sesuai permintaan. Data yang ditampilkan adalah node dari hasil pengambilan data berdasarkan XPath. Jika format kalimat tidak sesuai dengan aturan maka yang dimunculkan oleh aplikasi ini adalah sebuah peringatan kesalahan. Sedangkan jika input kalimat yang diberikan sudah sesuai dengan format aturan, tetapi data tidak tersedia maka aplikasi ini akan memunculkan sebuah tabel kosong. Prinsip kerja dari aplikasi ini adalah :

1. Menerima input berupa kalimat. Kalimat dengan format yang benar maupun format yang keliru akan tetap menghasilkan tampilan.
2. Kalimat input akan dipisahkan menjadi kata – kata. Pemisahan hanya berdasarkan urutan kata yang diinputkan.
3. Dari hasil pemisahan kalimat akan muncul kumpulan kata. Kata – kata yang muncul akan dibandingkan dengan kata – kata yang sudah terdaftar.

4. Perbandingan kata dengan daftar kata akan memunculkan dua kelompok kata yaitu kelompok kata yang bermakna (token) dan kelompok kata yang tidak mempunyai makna.
5. Kata yang tidak bermakna akan disingkirkan dan tidak dilakukan proses terhadapnya. Sedangkan kata yang bermakna akan dibandingkan dengan daftar kata yang termasuk atribut, pelengkap, operator dan kondisi.
6. Dari kumpulan kata – kata yang penting, akan dikonstruksikan menjadi XPath sesuai dengan aturan produksi pembentuk kalimat. Jika sudah sesuai dengan aturan produksi, maka akan diterjemahkan dalam XQuery untuk mengakses basisdata XML, sehingga menghasilkan keluaran field yang sesuai dengan node yang dituju.
7. Keluaran yang dihasilkan adalah sebuah tabel yang muncul pada sebuah form. Tabel tersebut berisi tampilan data perundangan yang sesuai dengan apa yang ditanyakan atau diperintahkan oleh user.

5.2. Bagian – Bagian Sistem

5.2.1. Scanner

Proses scanning berfungsi untuk memilah kata yang penting dan tidak penting. Kata yang penting akan disimpan, sedangkan kata yang tidak penting akan diabaikan. Langkah – langkah proses scanning adalah sebagai berikut :

1. Menerima kalimat masukan
2. Membentuk kalimat ke dalam list kata – kata
3. Dari hasil list yang didapat, maka karakter atau kata yang tidak berarti dapat diabaikan. (Wibisono, 2008)

5.2.2. Parser

Pada perancangan query Bahasa Indonesia ini, parser melakukan pelacakan masukan kalimat untuk mendapatkan langkah penyusunan kalimat yang tepat. Langkah ini dilakukan dari simbol awal sampai ke kalimat yang dihasilkan. Langkah – langkah dalam proses scanning adalah sebagai berikut :

1. Membaca daftar kata hasil scanning yang sudah terseleksi
2. Menentukan frase atribut

3. Memeriksa apakah ada ekor atribut pada frase atribut
4. Menentukan frase kondisi

Teknik yang digunakan dalam parsing adalah perbedaan list. Perbedaan list awal dengan ekor list akan menyisakan satu kata, kemudian kata tersebut diperiksa posisinya dalam query sesuai dengan aturan produksi untuk menentukan apakah termasuk frase atribut atau frase kondisi, apakah atribut atau konstanta. (Wibisono, 2008)

5.2.3 Translator

Translator berfungsi untuk mengubah pohon sintaks hasil parsing ke tipe – tipe query yang sesuai. Aturan produksi yang ditentukan dapat menghasilkan kalimat yang kompleks, karena frase atribut dapat memiliki satu ekor atribut dan frase kondisi dapat mempunyai nol atau satu ekor frase kondisi. Translator berfungsi untuk mengubah pohon sintaks yang susunan frase atribut dan frase kondisinya sesuai dengan pola pertanyaan – pertanyaan.

Langkah – langkah proses penerjemahan dalam translator adalah sebagai berikut :

1. Membaca pohon sintaks hasil parsing
2. Menempatkan atribut dari frase atribut ke posisi atribut pertama dalam notasi tipe query
3. Menempatkan atribut dan konstanta (data, int, opr) dari frase kondisi ke dalam notasi tipe query sesuai posisinya. (Wibisono, 2008)

5.2.4. Evaluator

Evaluator berfungsi menentukan jawaban akhir query berdasarkan hasil keluaran translator yang berupa penggolongan query menurut tipenya dengan urutan atribut sesuai dengan notasi masing –masing tipe query.

Proses yang terjadi dalam evaluator adalah pencarian relasi yang terdapat dalam query dan pencarian nilai yang tepat dalam basisdata sesuai dengan relasi tersebut. Setiap tipe query mempunyai urutan atribut yang berbeda yang akan menentukan proses evaluasinya. Dengan demikian setiap tipe query mempunyai proses evaluasi yang unik. Satu tipe query dapat mempunyai beberapa macam evaluasi. (Wibisono, 2008)

5.3. Struktur Data

Dalam proses koleksi data maka langkah awal yang dilakukan adalah penyimpanan data dengan format XML. Data perundangan dibuat dalam XML schema hal ini dimaksudkan untuk memastikan bahwa elemen – elemen dan atribut – atribut yang dimasukkan ke dalam dokumen sudah memenuhi aturan yang diterapkan dalam skema. XML Schema TblPerundangan untuk data yang disimpan adalah sebagai berikut :

1. jenisperundangan
2. nomor
3. tahun
4. isi :
 - a. namaperundangan
 - b. pejabat pembuat
 - c. penyetuju1
 - d. penyetuju2
 - e. disahkan
 - f. diundangkan

Penggambaran data dengan format XML pada XML Schema akan terlihat seperti gambar 5.2. Field yang dimiliki dokumen perundangan akan diperlihatkan sebagai field utama dengan tipe data string, sedangkan field yang dimiliki oleh isi diperlihatkan sebagai sebuah susunan yang bersarang dengan tipe data string.

```

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!-- W3C Schema generated by XMLSpy v2005.0 (http://www.xmlspy.com) -->
- <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:element name="disahkan" type="xs:string" />
  <xs:element name="diundangkan" type="xs:string" />
  <xs:element name="dokumen">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="jenisperundangan" />
        <xs:element ref="nomor" />
        <xs:element ref="tahun" />
        <xs:element ref="isi" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="isi">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="namaperundangan" />
        <xs:element ref="pejabatpembuat" />
        <xs:element ref="penyetuju1" />
        <xs:element ref="penyetuju2" />
        <xs:element ref="disahkan" />
        <xs:element ref="diundangkan" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="jenisperundangan" type="xs:string" />
  <xs:element name="namaperundangan" type="xs:string" />
  <xs:element name="nomor" type="xs:string" />
  <xs:element name="pejabatpembuat" type="xs:string" />
  <xs:element name="penyetuju1" type="xs:string" />
  <xs:element name="penyetuju2" type="xs:string" />
  <xs:element name="tahun" type="xs:string" />
</xs:schema>

```

Gambar 5.2. XML Schema

Penggambaran data dengan format XML pada XML Grid akan terlihat seperti gambar 5.3. Field yang dimiliki dokumen akan diperlihatkan sebagai elemen utama, sedangkan field yang dimiliki oleh isi diperlihatkan sebagai elemen dengan susunan yang bersarang.

- dokumen	
xmlns:xs	http://www.w3.org/2001/XMLSchema-instance
xmlns:mallemespa...	PELITAN PEJABATANPELITANPELITAN 2011PerundangSkema.xsd
jenisperundang...	peraturan daerah
nomor	5
tahun	2006
- isi	
namaperunden...	perubahan daerah rural perubahan hewan dan budidaya hewan potong kota serang
pejabatpembuat	walikota serang
penyetuju1	deputi kota serang
penyetuju2	walikota serang
disahkan	5 Juli 2006
diundangkan	3 desember 2007

Gambar 5.3. XML Grid

5.4. Identifikasi Tipe – Tipe Query

Penentuan pola pertanyaan dan pernyataan dalam memberikan masukan akan sangat menentukan dalam penggolongan tipe query. Simbol – simbol yang digunakan dalam penulisan pola masukan adalah sebagai berikut :

1. [T/P] : Kata tanya atau kata perintah yang mengawali masukan.
2. [Plk] : Kata pelengkap, terdiri dari satu atau lebih kata pelengkap.
3. <atribut> : Kata yang termasuk dalam kategori atribut, sesuai dengan daftar kata yang terdapat dalam daftar token.
4. <opr_bukan> : Kata – kata yang termasuk dalam kategori operator bukan, atau kata yang mempunyai arti sama dengan “bukan”
5. <opr> : simbol atau kata – kata yang berfungsi sebagai operator dapat berupa kata “sebelum” dan “sesudah” “lebih” dan “kurang”
6. <ktsb> : simbol atau kata – kata yang berfungsi sebagai kata sambung, yaitu tanda koma (,) serta kata sambung “dan”.
7. <atribut_kondisi> : simbol atau kata – kata yang berupa atribut, tetapi berfungsi sebagai atribut kondisi
8. <data> : tergantung dari data yang ada pada query

(Hartati dan Zuliarso, 2008)

Ketujuh tipe query yang diidentifikasi adalah sebagai berikut :

1. Tipe q_a (query – atribut)

Tipe query ini hanya berisi satu atribut pada kalimat yang berfungsi sebagai pertanyaan atau pernyataan. Atribut itulah yang akan ditampilkan. Query ini merupakan tipe yang paling sederhana karena hanya memuat atribut yang ditanyakan.

Tipe ini mempunyai pola masukan, yaitu :

- a. [T/P] <atribut> [Plk]

Contoh : Tampilkan mata kuliah

- b. [T/P] [Plk] <atribut>

Contoh : Tampilkan daftar soal

2. Tipe q_a_a (query – atribut – atribut)

Tipe query ini berisi beberapa atribut yang akan ditampilkan. Untuk memisahkan satu atribut dengan atribut berikutnya digunakan kata sambung ‘dan’ atau tanda baca koma ‘,’.

Tipe ini mempunyai pola masukan, yaitu :

a. [T/P] <atribut> (<ktsb> <atribut>)*

Contoh : Tampilkan kode, mata kuliah , no , soal dan bobot

b. ([T/P] [Plk]* <atribut>)*

Contoh : Apa soal dan berapa bobotnya ?

3. Tipe q_a_opr (query – atribut – operator)

Tipe query berisi satu atribut yang akan ditampilkan dan satu kondisi. Tipe ini mempunyai pola masukan, yaitu :

[T/P] <atribut> [Plk] <atribut> <data>

Contoh : Cari soal yang mengandung kata pendengaran

4. Tipe q_a_a_opr (query – atribut – atribut – operator – atribut – operator)

Tipe query berisi beberapa atribut yang akan ditampilkan dan beberapa kondisi. Tipe query ini memuat beberapa atribut yang ditanyakan. Untuk memisahkan satu atribut dengan atribut berikutnya digunakan kata sambung ‘dan’ atau tanda baca koma ‘,’. Demikian juga untuk memisahkan satu atribut kondisi dengan atribut kondisi berikutnya digunakan kata sambung ‘dan’ atau tanda baca koma ‘,’.

Tipe ini mempunyai pola masukan, yaitu :

[T/P] <atribut> (<ktsb> <atribut>)* [Plk] <atribut> <data> <ktsb>

<atribut><data>

Contoh :

Tampilkan kode, mata kuliah , hari, jam , tanggal dengan program studi mengandung kata sistem dan mata kuliah mengandung kata sistem

5. Tipe q_a_opr_data (query – atribut – operator – < data>)

Tipe query berisi beberapa atribut yang akan ditampilkan dan kondisi operator ‘lebih’, ‘kurang’, ‘sebelum’ atau ‘sesudah’.

Tipe ini mempunyai pola masukan, yaitu :

[T/P] <atribut> (<ktsb> <atribut>)* [Plk] <atribut><opr><data>

Contoh :

Tampilkan kode, mata kuliah, progdi , semester, no , soal dengan no > 6

6. Tipe q_a_bukan (query – atribut – bukan – data)

Tipe query berisi beberapa atribut yang akan ditampilkan dan kondisi operator “bukan” atau “tidak” atau “selain”.

Tipe ini mempunyai pola masukan, yaitu :

[T/P] <atribut> (<ktsb> <atribut>)* [Plk] <atribut><opr_bukan><data>

Contoh :

Tampilkan matakuliah dengan kode bukan P

7. Tipe q_a_bukan (query – atribut – atribut – bukan – data)

Tipe query berisi beberapa atribut yang akan ditampilkan dan kondisi operator “bukan” atau “tidak” atau “selain”.

Tipe ini mempunyai pola masukan, yaitu :

[T/P] <atribut> (<ktsb> <atribut><atribut><op><data>)* [Plk]
<atribut><opr_bukan><data>

Contoh :

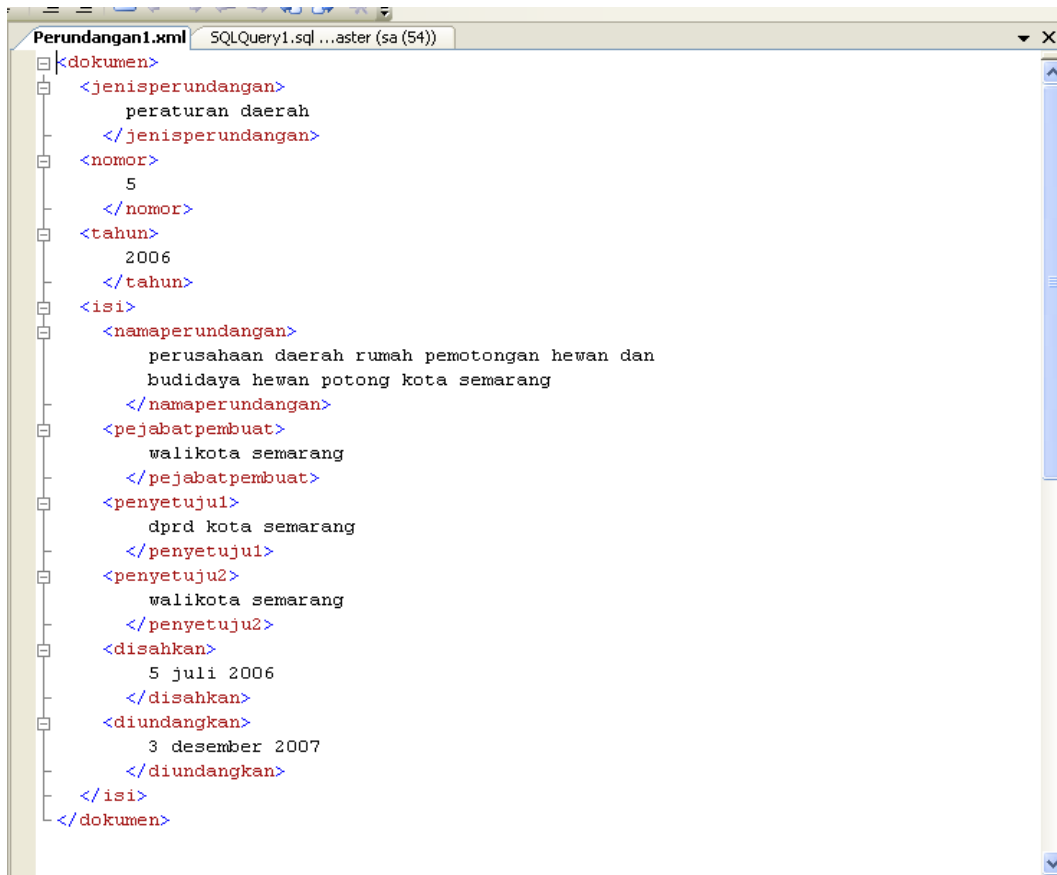
Cari kode dan mata kuliah dengan kode bukan A

(Hartati dan Zuliarso, 2008)

5.5. Implementasi Basisdata

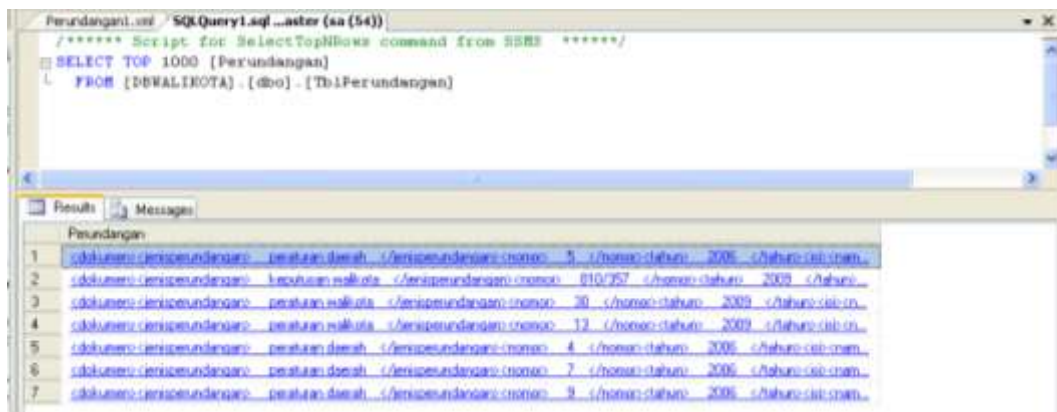
Ketika basisdata produk perundangan Pemerintah Kota Semarang akan digunakan, maka harus dibuat terlebih dahulu. Sehingga langkah awal dalam implementasi sistem ini adalah merancang dan mengimplementasikan basisdata perundangan Pemerintah Kota Semarang sebagai basisdata XML terlebih dahulu.

Data produk perundangan Pemerintah Kota Semarang ditampilkan dalam struktur data seperti ditampilkan dalam gambar 5.4.



Gambar 5.4. Struktur data Perundangan Pemerintah Kota Semarang dalam format XML

Data perundangan Pemerintah Kota Semarang dalam format XML diimplementasikan menggunakan SQL Server 2008. Gambar 4.2 adalah tampilan data perundangan Pemerintah Kota Semarang dalam format XML.



Gambar 5.5. Tampilan data perundangan dalam format XML

5.6. Implementasi Komponen Pemroses Bahasa

5.6.1. DisplayQueryNlp.java

Pada subprogram ini ditentukan koneksi aplikasi dengan database dan tabel yang digunakan dalam aplikasi ini. Database yang digunakan adalah "DBWALIKOTA" dan tabel yang digunakan adalah "TblPerundangan", dengan username "sa" dan password "aliya", seperti terlihat pada gambar 5.6

```
public class DisplayQueryNlp extends JFrame
{
    // JDBC driver and database URL
    static final String JDBC_DRIVER      = "sun.jdbc.odbc.JdbcOdbcDriver";
    //static final String DATABASE_URL    = "jdbc:odbc:TblPerundangan";
    static final String DATABASE_URL    = "jdbc:odbc:DBWALIKOTA";
    static final String USERNAME        = "sa";
    static final String PASSWORD        = "aliya";

    // default query selects all rows from authors table
    static final String DEFAULT_QUERY   = "select Perundangan.value('/dokumen[1]/jenisperundangan[1]

    private ResultSetTableModel tableModel;
    private JTextArea queryArea;

    // create ResultSetTableModel and GUI
    public DisplayQueryNlp()
    {
        super ( "AKSES NODE XPATH " );

        // create ResultSetTableModel and display database table
        try
        {
            // create TableModel for results of query SELECT * FROM authors
```

Gambar 5.6. Koneksi database

Ketika aplikasi dijalankan akan ditampilkan sebuah default query bahasa Indonesia yang sesuai dengan format kalimat. Jika format query sudah benar, tetapi data yang dikehendaki tidak tersedia, maka akan ditampilkan tabel kosong. Begitu juga jika terdapat kesalahan dalam penulisan format query, maka akan ditampilkan pesan kesalahan, seperti terlihat pada gambar 5.7.

```
catch ( SQLException sqlException )
{
    JOptionPane.showMessageDialog( null,

    //sqlException.getErrorCode(),
    sqlException.getMessage(),
    "Database Error Query Tidak berhasil ",
    JOptionPane.ERROR_MESSAGE );

    // try to recover from invalid user query
    // by executing default query
    try
    {
        tableModel.setQuery( DEFAULT_QUERY );
        queryArea.setText( DEFAULT_QUERY );
    } // end try
    catch ( SQLException sqlException2 )
    {
        JOptionPane.showMessageDialog( null,
        sqlException2.getMessage(), "Database Error ",
        JOptionPane.ERROR_MESSAGE );
```

Gambar 5.7. Pesan kesalahan

5.6.2. Scanner

Dalam implementasi sistem ini, scanner yang digunakan adalah utilitas scanner yang ada dalam library Java. Untuk menggunakannya cukup dengan memanggilnya, dengan proses pemanggilan diimplementasikan seperti terlihat pada gambar 4.7.

```
package nlp.parser;
import java.util.Scanner;
import java.util.regex.Pattern;
```

Gambar 5.8. Penggunaan Utility Scanner

5.6.3. Parser

Analisis sintaks dilakukan berdasarkan aturan produksi yang telah ditentukan dan akan menghasilkan pohon sintaks (pohon parser). Seperti terlihat pada gambar 5.9. proses pengubahan pohon sintaks hasil parsing ke dalam tipe – tipe query diimplementasikan dalam bagian program ini.

```
public class Parsing
{
    public String Pengurai(String str)
    {
        String names[] = { "jenis", "jenisperundangan", "nama", "namaperundangan", "pejabat", "pejabatpeml
        SortedSet< String > treex = new TreeSet< String >( Arrays.asList( names ) );

        List list
        List listkon
        List listekor
        String tdan
        String koma
        boolean frasekondisi
        boolean adaekor
        String tok, tokenlist;
        String bantu
        String bantuan
        String kkatakunci
        StringBuffer sbf
        StringBuffer sbfbantu

        String regex = "|dimana|apa|saja|tampilkan|tampil|cari|daftar|mengandung|di|pada|dari|yang|o.
```

Gambar 5.9. Deklarasi Parser

5.6.4. Translator

Pohon sintaks yang dihasilkan dari proses parsing kemudian diterima translator. Translator berfungsi menghasilkan kode antara, sebelum memasuki proses evaluasi untuk menentukan jawaban akhir query. Komponen evaluator berfungsi mengevaluasi daftar *goal* dalam proses penentuan jawaban query, sehingga evaluasi banyak terlibat dengan entri – entri basisdata, seperti terlihat pada gambar 5.10.

```

public class Evaluator
{
    private List fraseatribut;
    private List frasekondisi;
    private StringBuffer strbf=new StringBuffer("select distinct ");
    private Object bantu,bantuan;

    // Variabel bersama querysatu dan querydua
    String fdan = "dan";
    String fatau = "atau";
    String fcomma = ",";
    String sambung[] = {"dan",","," "};
    StringBuffer strbff = new StringBuffer("");
    String ekorkon = " ";
    String ekorkonp = " "; //ekor query mahasiswa
    String utama[] = {"jenisperundangan","nomor","tahun"};
    String isi[] = {"namaperundangan","pejabatpembuat","penyetujui","penyetuju2","disahkan","di

    SortedSet< String > treeutama = new TreeSet< String >( Arrays.asList( utama ) );
    SortedSet< String > treeisi = new TreeSet< String >( Arrays.asList( isi ) );
    SortedSet< String > treesambung = new TreeSet< String >( Arrays.asList( sambung ) );

    // Variabel Khusus querydua
    String fkurang = "kurang";
    String flebih = "lebih";
    String suka = "like"; //jika ada kata bukan/tidak
    String sukata = "=";
    Boolean statkon = false; //untuk status apakah di atribut kondisi ada dan
    String tolak[] = {"bukan","tidak"};
    String danatau[] = {"atau","dan"};
    SortedSet< String > treetolak = new TreeSet< String >( Arrays.asList( tolak ) );
    SortedSet< String > treedanatau = new TreeSet< String >( Arrays.asList( danatau ) );
    int posisi;

```

Gambar 5.10. Translator

5.6.5. Evaluator

Baris pertama akan mengambil satu simpul terdepan dari linked list token untuk dimasukkan ke dalam variabel bantu. Baris kedua akan menguji apakah himpunan treeutama mengandung string yang dimuat dalam variabel bantu. Jika ya, maka kemudian dilakukan penterjemahan ke dalam fragmen sintaks XQuery.

Untuk XML dengan data bersarang, proses pengubahan pohon sintaks hasil parsing ke dalam tipe – tipe query diimplementasikan dalam bagian program seperti terlihat pada gambar 4.14. Setelah dilakukan penguraian berdasarkan frase atribut dan frase kondisi. Selanjutnya dilakukan penguraian berdasarkan masing – masing frase.

```

//-----
// Mulai querysatu
public String querysatu(List fratribut,List frakondisi)
{
    while (fratribut.isEmpty() != true)
    {
        bantu=fratribut.removeFromFront();
        if (treeutama.contains(bantu)) //atribut kode, matakuliah, progdi dll
        {
            strbf.append(" Perundangan.value('/dokumen[1]/")
            .append(bantu)
            .append("[1]','varchar(200)') as ")
            .append(bantu);
            continue;
        }

        if (treeisi.contains(bantu)) // atribut deskripsi
        {
            strbf.append("a.value('");
            strbf.append(bantu)
            .append("[1]','varchar(200)') as ")
            .append(bantu);
            ekorkonp="CROSS APPLY Perundangan.nodes('/dokumen/isi') as result(a)";
            continue;
        } // atribut deskripsi
    }
}

```

Gambar 5.11. Evaluator

5.6.6. DemoFrame.java

Program DemoFrame ini digunakan untuk memperlihatkan pola pertanyaan sekaligus beserta contoh kalimat masukan yang dapat memberikan jawaban dengan benar, seperti terlihat pada gambar 5.12.

```

public DemoFrame()
{
    queryArea = new JTextArea
    (
        "POLA PERNYATAAN INPUT BESERTA CONTOH \n \n" +
        "----- \n \n" +
        "POLA 1 : QUERY - ATRIBUT \n" +
        "        Tampilkan jenis perundangan \n \n" +
        "POLA 2 : QUERY - ATRIBUT - ATRIBUT \n" +
        "        Tampilkan jenis perundangan , tahun \n \n" +
        "POLA 3 : QUERY - ATRIBUT - ATRIBUT - OPERATOR \n" +
        "        Tampilkan jenis perundangan , nomor dan tahun dengan tahun 2006 \n \n" +
        "POLA 4 : QUERY - ATRIBUT - ATRIBUT - OPERATOR - ATRIBUT - OPERATOR \n" +
        "        Tampilkan jenis perundangan , nomor dengan pejabat pembuat walikota dan tahun dengan
        "POLA 5 : QUERY - ATRIBUT - OPERATOR - DATA \n" +
        "        Tampilkan jenis perundangan , nomor dan tahun dengan tahun lebih dari 2006\n \n" +
        "POLA 6 : QUERY - ATRIBUT - BUKAN - DATA \n" +
        "        Tampilkan jenis perundangan dengan tahun bukan 2006 \n \n" +
        "POLA 7 : QUERY - ATRIBUT - ATRIBUT - BUKAN - DATA \n" +
        "        Tampilkan jenis perundangan , nomor dan tahun dengan tahun bukan 2006 \n \n" +
        "----- \n \n"
        , 20, 300 );
    queryArea.setWrapStyleWord( true );
    queryArea.setLineWrap( true );
    queryArea.setEditable(false);
}

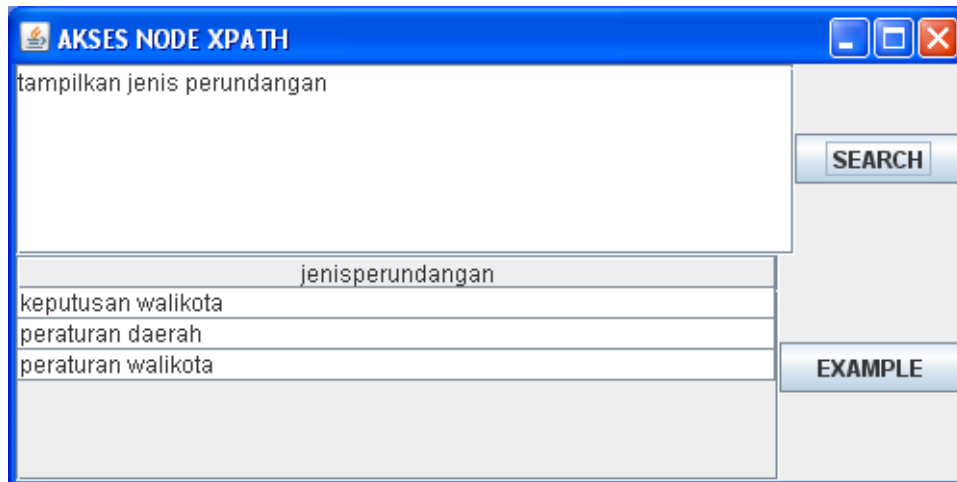
public void createAndShowGUI()
{
    JFrame.setDefaultLookAndFeelDecorated(true);
    JFrame frame = new JFrame("BEBERAPA POLA DAN CONTOH PERNYATAAN INPUT");
    frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
    frame.getContentPane().add(queryArea);
}

```

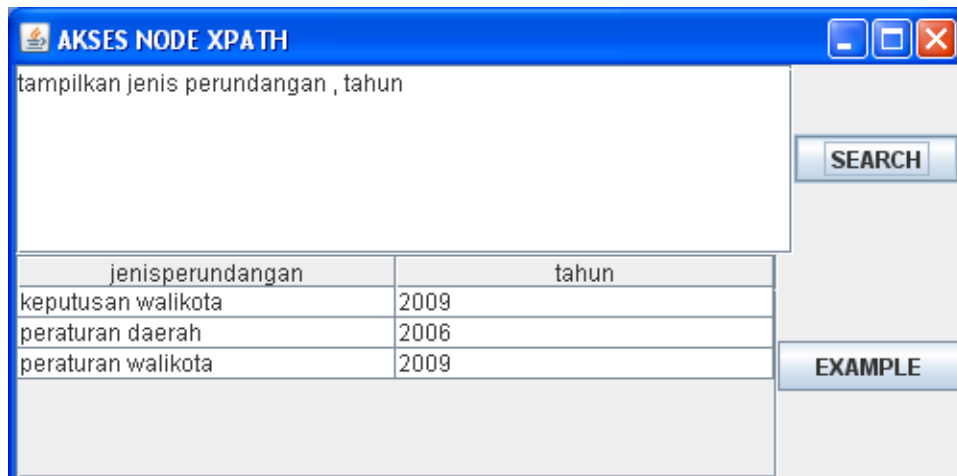
Gambar 5.12. DemoFrame

5.7. Tampilan Aplikasi

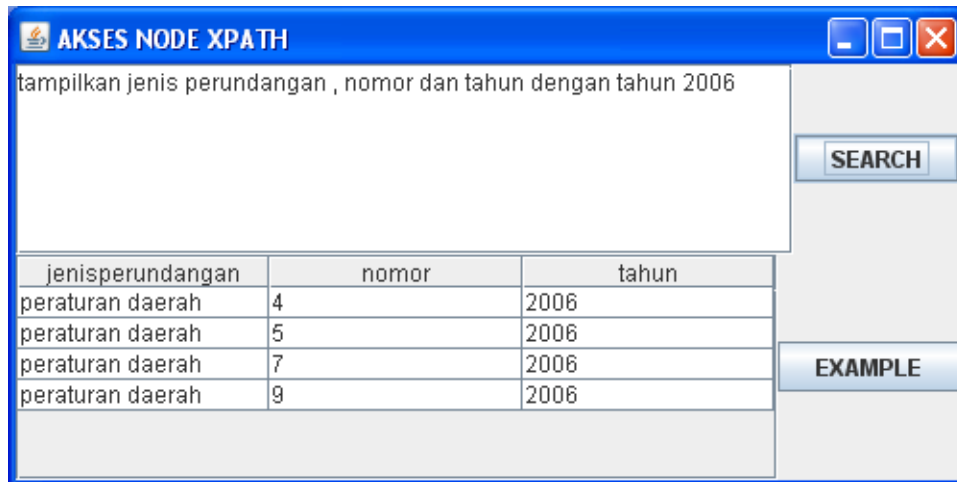
5.7.1. Tampilan Tipe q_a (query – atribut)



5.7.2. Tampilan Tipe q_a_a (query – atribut – atribut)



5.7.3. Tampilan Tipe q_a_opr (query – atribut – operator)



The screenshot shows a window titled "AKSES NODE XPATH" with a search query: "tampilkan jenis perundangan , nomor dan tahun dengan tahun 2006". Below the query is a table with three columns: "jenisperundangan", "nomor", and "tahun". The table contains four rows of data. To the right of the table are two buttons: "SEARCH" and "EXAMPLE".

jenisperundangan	nomor	tahun
peraturan daerah	4	2006
peraturan daerah	5	2006
peraturan daerah	7	2006
peraturan daerah	9	2006

5.7.4. Tampilan Tipe q_a_a_opr (query – atribut – atribut – operator – atribut – operator)



The screenshot shows a window titled "AKSES NODE XPATH" with a search query: "tampilkan jenis perundangan , nomor dan tahun". Below the query is a table with three columns: "jenisperundangan", "nomor", and "tahun". The table contains six rows of data. To the right of the table are two buttons: "SEARCH" and "EXAMPLE".

jenisperundangan	nomor	tahun
keputusan walikota	810/357	2009
peraturan daerah	4	2006
peraturan daerah	5	2006
peraturan daerah	7	2006
peraturan daerah	9	2006
peraturan walikota	13	2009

5.7.5. Tampilan Tipe q_a_bukan (query – atribut – bukan – data)

The screenshot shows a window titled "AKSES NODE XPATH" with a search bar containing the text "Tampilkan jenis perundangan , nomor dan tahun dengan tahun bukan 2006". A "SEARCH" button is located to the right of the search bar. Below the search bar is a table with three columns: "jenisperundangan", "nomor", and "tahun". The table contains three rows of data. To the right of the table is an "EXAMPLE" button.

jenisperundangan	nomor	tahun
keputusan walikota	810/357	2009
peraturan walikota	13	2009
peraturan walikota	30	2009

5.7.6. Tampilan contoh-contoh format kalimat input

The screenshot shows a window titled "CONTOH PERTANYAAN" with a list of query patterns. The window title bar includes a folder icon and window control buttons. The content is as follows:

POLA PERTANYAAN BESERTA CONTOH

POLA 1 : QUERY - ATRIBUT
Tampilkan jenis perundangan

POLA 2 : QUERY - ATRIBUT - ATRIBUT
Tampilkan jenis perundangan , tahun

POLA 3 : QUERY - ATRIBUT - ATRIBUT - OPERATOR
Tampilkan jenis perundangan , nomor dan tahun dengan tahun 2006

POLA 4 : QUERY - ATRIBUT - ATRIBUT - OPERATOR - ATRIBUT - OPERATOR
Tampilkan jenis perundangan , nomor dengan pejabat pembuat walikota dan tahun dengan tahun 2006

POLA 5 : QUERY - ATRIBUT - OPERATOR - DATA
Tampilkan jenis perundangan , nomor dan tahun dengan tahun lebih dari 2006

POLA 6 : QUERY - ATRIBUT - BUKAN - DATA
Tampilkan jenis perundangan dengan tahun bukan 2006

POLA 7 : QUERY - ATRIBUT - ATRIBUT - BUKAN - DATA
Tampilkan jenis perundangan , nomor dan tahun dengan tahun bukan 2006

BAB VI

KESIMPULAN DAN SARAN

6.1. Kesimpulan

Berdasarkan hasil penelitian ini, maka dapat disimpulkan beberapa hal sebagai berikut :

1. Aplikasi ini dapat digunakan sebagai mesin pencari untuk perolehan data produk perundangan Pemerintah Kota Semarang
2. Untuk memperoleh informasi, aplikasi ini menggunakan XPath untuk mendapatkan node atau data yang sesuai dengan permintaan. XPath tidak secara eksplisit terlihat dalam listing program maupun dalam tampilan, tetapi XPath diletakkan pada bagian program Translator. Pada Translator itulah sebenarnya peran XPath dalam mengakses node data.
3. Aplikasi ini memberikan contoh pertanyaan maupun perintah untuk mengakses node.
4. Aplikasi ini mampu memberikan keluaran berupa tabel yang berisi data. aplikasi ini juga mampu menampilkan XQuery hasil translasi dari kalimat masukan, dalam hal ini berupa pernyataan SQL dalam format XML.

6.2 Saran

1. Pernyataan XPath untuk mengakses node dalam bentuk XQuery diperlukan agar dapat memberikan petunjuk ketika user melakukan kesalahan input perintah. Sehingga peringatan kesalahan terhadap kesalahan akses basisdata dapat ditampilkan dalam bentuk peringatan yang berupa XPath yang langsung berkaitan dengan kesalahan akses data.
2. Diperlukan suatu mekanisme tambahan berupa contoh pernyataan maupun pertanyaan input yang keliru, sehingga pengguna tidak akan terlalu bingung ketika input yang diberikan tidak memberikan hasil sesuai yang dikehendaki.
3. Dengan format pertanyaan maupun pernyataan input yang masih terbatas, maka perlu untuk membuat lebih banyak variasi dalam memberikan input, sehingga pengguna mempunyai banyak pilihan variasi dalam pencarian data.

DAFTAR PUSTAKA

Elliott Rusty Harold, E.R. dan Means, W.S., 2001, “*XML in a Nutshell A Desktop Quick Reference*”, <http://oreilly.com>

Hartati, S., dan Zuliarso E., 2008, *Aplikasi Pengolah Bahasa Alami untuk Query Basisdata XML*, Dinamik, Vol XIII, No 2, Juli 2008, 168 – 175.

Murti, M.S., 2010, ” *Harmonisasi Peraturan Daerah Dengan Peraturan Perundang Undangan Lainnya*”, www.djpp.depkuham.go.id

Wibisono, S., 2008, “*Aplikasi Pengolah Bahasa Alami untuk Query Basisdata Akademik dengan Format Data XML*”, Tesis Program S2 Ilmu Komputer, Universitas Gadjah Mada

www.rpbouret.com/xml/XMLAndDatabases.htm/#isxmladatabase

www.w3.org

www.rpbouret.com/xml/XMLAndDatabases.htm/#xmlquery

DAFTAR RIWAYAT PENELITI (KETUA)

I. DATA DIRI

1. Nama Lengkap : Imam Husni Al Amin, ST
2. NIY : YU.2.04.04.068
3. Golongan/Pangkat : IIIC / Penata
4. Jabatan Fungsional : Lektor
5. Tempat, Tgl. Lahir : Magelang, 28 April 1973
6. Jenis Kelamin : Laki-laki
7. Alamat Rumah : Jl. Gemah Selatan II No 56 Pedurungan
8. Telp / Faks / e-mail : 081325438866 / - / sonny@unisbank.ac.id
9. Alamat Kampus : Jl. Trilomba Juang 1 Semarang
10. Telp / Faks / e-mail : 8311668 / 8443240 / info@unisbank.ac.id

II. RIWAYAT PENELITIAN

<u>No</u>	<u>Judul</u>	<u>Tahun</u>	<u>Keterangan</u>
1.	Rancang Bangun Web Pembinaan Jarak Jauh (PJJ) ONN Bidang Komputer Untuk SMA/SMK Di Jawa Tengah	2010	Anggota
2.	Pengendalian Kualitas Pada Industri Manufaktur Dengan Menggunakan Statistical Process Chart	2008	Ketua
3.	Peningkatan Efisiensi Penggunaan Tenaga Kerja Pada Sektor Industri Kecil Di Jawa Tengah.	2006	Anggota
4.	Pengaruh Permodalan Terhadap Peningkatan Rentabilitas Perusahaan	2003	Anggota

Demikian daftar riwayat penelitian ini dibuat dengan sebenarnya.

Semarang, 15 Desember 2011

Imam Husni Al Amin, ST

DAFTAR RIWAYAT PENELITI (ANGGOTA)

I. DATA DIRI

1. Nama Lengkap : Fatkhul Amin, ST
2. NIY : YU.2.02.10.044
3. Golongan/Pangkat : IIIB/ Penata Muda Tk.I
4. Jabatan Fungsional : Asisten Ahli
5. Tempat, Tgl. Lahir : Pekalongan, 24 September 1974
6. Jenis Kelamin : Laki-laki
7. Alamat Rumah : Jl. Candi Pawon Timur VI no 15,Manyaran, Smg
8. Telp / Faks / e-mail : fatkhulamin@gmail.com
9. Alamat Kampus : Jl. Trilomba Juang 1 Semarang
10. Telp / Faks / e-mail : 8311668 / 8443240 / info@unisbank.ac.id

II. RIWAYAT PENELITIAN

No	Judul	Tahun	Keterangan
1.	Peningkatan Efisiensi Penggunaan Tenaga Kerja Pada Sektor Industri Kecil Di Jawa Tengah.	2010	Anggota
2.	Capital Budgeting Sebagai Alat Penilaian Kebijakan Investasi Aktiva Tetap Pada Perusahaan Batik Aini	2006	Ketua
3.	Metode Perkiraan Laju Aliran Puncak (Debit Air) Sebagai Dasar Analisis Sistem Drainase Di Daerah Aliran Sungai Wilayah Semarang Berbantuan Sig	2005	Ketua

Demikian daftar riwayat penelitian ini dibuat dengan sebenarnya.

Semarang, 15 Desember 2011

Fatkhul Amin, ST

DAFTAR RIWAYAT PENELITI (ANGGOTA)

I. DATA DIRI

1. Nama Lengkap : Setyawan Wibisono, S.Kom., M.Cs
2. NIY : 197306072005011001
3. Golongan/Pangkat : IIIB/ Penata Muda Tk.I
4. Jabatan Fungsional : Lektor
5. Tempat, Tgl. Lahir : Ambarawa, 7 Juni 1973
6. Jenis Kelamin : Laki-laki
7. Alamat Rumah : Jl. Sinar Gemah Timur 964H, Perum Sinar Waluyo,
Semarang
8. Telp / Faks / e-mail : 081325438866 /-/ sonny@unisbank.ac.id
9. Alamat Kampus : Jl. Trilomba Juang 1 Semarang
10. Telp / Faks / e-mail : 8311668 / 8443240 / info@unisbank.ac.id

II. RIWAYAT PENELITIAN

No	Judul	Tahun	Keterangan
1.	Query Bahasa Indonesia Pada Basis Data Soal Ujian Di Fakultas Teknologi Informasi Universitas Stikubank Dengan Format Data XML	2011	Anggota
2.	Aplikasi Generator Nilai Tag Meta (Meta Tag Value) Pada Halaman Web Untuk Meningkatkan Peringkat Temu Kembali Informasi Mesin Pencari	2011	Anggota
3.	Aplikasi Pengelolaan Peraturan Daerah Provinsi Jawa Tengah Menggunakan Basis Data XML	2010	Ketua
4.	Adaptive Tutoring System Untuk Pembelajaran Bahasa Pemrograman Di Universitas Stikubank Semarang	2010	Anggota

Demikian daftar riwayat penelitian ini dibuat dengan sebenarnya.

Semarang, 15 Desember 2011

Setyawan Wibisono, S.Kom., M.Cs

DAFTAR RIWAYAT PENELITI (MAHASISWA)

I. DATA DIRI

1. Nama Lengkap : Feri Dwi Astuti
2. NIM : 08.01.53.0056
3. Fakultas : Teknologi Informasi
4. Program Studi : Teknik Informatika
5. Tempat, Tgl. Lahir : Semarang, 25 Februari 1990
6. Jenis Kelamin : Perempuan
7. Alamat Rumah : DESA CAMPUREJO RT 04 RW 02 BOJA
8. Telp : 085740303017
9. E-mail : astuti.feri@gmail.com
10. Alamat Kampus : Jl. Trilomba Juang 1 Semarang
11. Telp / Faks / e-mail : 8311668 / 8443240 / info@unisbank.ac.id

II. RIWAYAT PENELITIAN

No.	Judul	Tahun	Keterangan

Demikian daftar riwayat penelitian ini dibuat dengan sebenarnya.

Semarang, 15 Desember 2011

Feri Dwi Astuti

DAFTAR RIWAYAT PENELITI (MAHASISWA)

I. DATA DIRI

1. Nama Lengkap : Agustina Wijaya
2. NIM : 08.01.53.0067
3. Fakultas : Teknologi Informasi
4. Program Studi : Teknik Informatika
5. Tempat, Tgl. Lahir : LAHAT, 04/08/1990
6. Jenis Kelamin : Perempuan
7. Alamat Rumah : Jl. Wahyu Asri VI/CIII RT 01 RW 06
8. Telp : 085640089271
9. e-mail : -
10. Alamat Kampus : Jl. Trilomba Juang 1 Semarang
11. Telp / Faks / e-mail : 8311668 / 8443240/ info@unisbank.ac.id

II. RIWAYAT PENELITIAN

No.	Judul	Tahun	Keterangan

Demikian daftar riwayat penelitian ini dibuat dengan sebenarnya.

Semarang, 16 Oktober 2011

Agustina Wijaya

JOB DESK

1. Ketua Peneliti

- a. Melakukan komunikasi dengan pihak Biro Hukum Pemerintah Kota Semarang untuk mendapatkan ijin riset, akses penggunaan data.
- b. Mengkoordinasi jadwal kegiatan.
- c. Mengkoordinasi penggunaan anggaran penelitian
- d. Melakukan penelitian seperti anggota peneliti.
- e. Melakukan pembuatan proposal dan laporan.

2. Anggota Peneliti

- a. Melakukan studi literasi.
- b. Melakukan komunikasi dengan pihak Biro Hukum Pemerintah Kota Semarang untuk menentukan penggunaan data perundangan.
- c. Melakukan koleksi data.
- d. Melakukan analisa kebutuhan basisdata.
- e. Melakukan rekayasa algoritma.
- f. Melakukan pembuatan program.
- g. Melakukan testing program
- h. Mencatat kegiatan perkembangan penelitian dalam log-book
- i. Mengkomunikasikan dengan ketua peneliti tentang kemajuan penelitian.
- j. Melakukan pembelanjaan dana penelitian.

REALISASI PENGGUNAAN ANGGARAN PENELITIAN

1. Bahan dan Peralatan Penelitian		
a. Bahan-bahan		
• Kertas HVS 3 rim @Rp.35.000	Rp	105.000
• Tinta Printer 1 Set @Rp 380.000	Rp.	375.000
• Modem	Rp	420.000
b. Flash disk 4 GB, 2 buah @Rp 90.000	Rp	180.000
c. External Hardisk 500 GB 1 buah @Rp 500.000	Rp	500.000
d. Line of code 350 baris @ Rp 2.000	Rp	700.000
e. Penggandaan dokumen 600 lembar @ Rp 150	Rp	90.000
		<hr/>
		Rp 2.370.000
2. Perjalanan dan Akomodasi		
a. Transportasi pengumpulan data dan informasi		
6 Hari @Rp. 15.000 untuk 2 orang	Rp	180.000
b. Akomodasi		
6 Hari @Rp. 15.000 untuk 2 orang	Rp	180.000
		<hr/>
		Rp 360.000
3. Seminar Penelitian		
a. Konsumsi 30 orang, @Rp 6.000	Rp	180.000
		<hr/>
		Rp 180.000
4. Pembuatan dan Penggandaan Laporan		
a. Fotokopi 60 lembar x 7 x Rp 150	Rp	63.000
b. Fotokopi 30 lembar x 2 x Rp 150	Rp	9.000
c. Jilid 9 Buah @ Rp 2.000,-	Rp	18.000
		<hr/>
		Rp 90.000,-
		<hr/>
TOTAL BIAYA	Rp.	3.000.000,00
		(Tiga juta rupiah)

LOKASI PENELITIAN

