

PENERAPAN TEKNOLOGI *SEMANTIC WEB* PADA APLIKASI PENCARIAN KOLEKSI PERPUSTAKAAN (STUDI KASUS:PERPUSTAKAAN FTI UPN "VETERAN" YOGYAKARTA)

Nava'atul Fadillah¹⁾, Novrido Charibaldi²⁾, Herlina Jayadianti³⁾

^{1,2,3)}Jurusan Teknik Informatika UPN "Veteran" Yogyakarta

Jl. Babarsari no 2 Tambakbayan 55281 Yogyakarta Telp (0274)-485323

e-mail: novrido@gmail.com, herlinajayadianti@gmail.com, navaatul@yahoo.com

Abstrak

Perpustakaan Fakultas Teknologi Industri UPN "Veteran" Yogyakarta menyimpan banyak koleksi pustaka seperti buku, jurnal, skripsi atau tesis, dan laporan kerja praktek, namun persoalan pun muncul dapatkah informasi pustaka tersebut dikelola secara semantis sehingga nantinya pengguna dapat melakukan pencarian berdasarkan persepsi dan tingkat pengetahuan yang dimilikinya untuk mencari koleksi pustaka. Dalam penelitian ini, pengetahuan koleksi perpustakaan disimpan dalam bentuk ontology berbasis Web Ontology Language yang memiliki class Jurusan, Koleksi, Penulis dan Penerbit. Model ontology yang dibangun diimplementasikan untuk aplikasi berbasis web menggunakan Protégé 3.4, NetBeans IDE 6.5.1 dengan bahasa pemrograman Java Server Pages (JSP), Jena API sebagai library pendukung, dan bahasa query SPARQL.

Keywords: *Ontology, Protégé 3.4, NetBeans IDE 6.5.1, JSP, Jena API, SPARQL.*

1. PENDAHULUAN

Hingga saat ini sistem yang digunakan untuk mengakses informasi buku di perpustakaan tersebut belum berbasis pengetahuan dan *searching* tidak dilakukan secara semantik, pencarian yang dihasilkan masih sangat terbatas dan belum sesuai dengan persepsi seseorang dalam mencari koleksi pada sebuah perpustakaan.

Persoalan menarik muncul adalah bagaimana menyimpan informasi tersebut secara lebih semantis sehingga dapat dilakukan pemanfaatan kembali seperti *query* yang disesuaikan dengan persepsi-persepsi dan tingkat pengetahuan masing-masing pengguna ketika akan mencari sebuah koleksi pada perpustakaan.

Salah satu upaya yang sekarang banyak dilakukan adalah dengan pendekatan model ontology. Ontologi mendukung suatu sistem *Knowledge Management* dan membuka kemungkinan untuk berpindah dari pandangan berorientasi dokumen ke arah pengetahuan yang saling terkait, dapat dikombinasikan, serta dapat dimanfaatkan kembali secara lebih fleksibel dan dinamis. Ontologi merupakan cara merepresentasikan pengetahuan tentang makna objek, properti dari suatu objek, serta relasi objek tersebut yang mungkin terjadi pada domain pengetahuan (Chandrasekaran, dan Josehson, 1999).

Tujuan penelitian ini adalah untuk menghasilkan suatu aplikasi dengan berbasis pengetahuan mengenai koleksi pada perpustakaan yang dituangkan dalam aplikasi web menggunakan *Ontology Web Language* sebagai bagian dari teknologi *Semantic Web* menggunakan *Java Sever Pages*, Jena, Protégé dan bahasa *query* SPARQL.

Manfaat Penelitian ini diharapkan dapat membantu *user* ketika akan mencari suatu koleksi di perpustakaan dengan mudah serta dapat mempermudah admin dalam pengelolaan data buku yang nantinya disimpan dalam protégé 3.4.

2. TINJAUAN PUSTAKA

Metodologi pengembangan perangkat lunak yang digunakan dalam penelitian ini adalah dengan menggunakan metode *waterfall* yaitu analisis, *design*, implementasi serta pengujian. Sebelumnya pernah dibuat aplikasi semacam itu dengan judul *knowledge management system* informasi tiket kereta api berbasis ontologi oleh Putri Taqwa (2008) dari Universitas Pembangunan Nasional "Veteran" Yogyakarta. Penelitian tersebut dilakukan dengan metodologi *ontological engineering* yaitu *basic topic, design, development, application, dan knowledge sharuing and reuse*.

Teknologi yang digunakan dalam penelitian ini dan yang dilakukan oleh Putri Taqwa dalam penelitiannya yaitu sama-sama menggunakan teknologi semantik. Namun yang membedakan penelitian ini dengan penelitian sebelumnya yaitu di dalam penelitian ini akan dibangun sebuah aplikasi yang akan menerapkan konsep semantik, sedangkan dalam penelitian sebelumnya sistem yang dibangun hanya sampai tahap pembangunan ontologi.

Kemudian bahasa pemrograman yang dipakai pada penelitian sebelumnya oleh Putri Taqwa yaitu menggunakan *ontology web language* (OWL), sedangkan pada penelitian ini selain menggunakan OWL juga menggunakan bahasa pemrograman JSP (*Java Server Pages*) untuk merancang *interface*-nya. Kemudian *tools* yang digunakan dalam penelitian sebelumnya yaitu Protégé 3.4, sedangkan pada penelitian ini *tools* yang

digunakan adalah Protégé 3.4 dan JENA API sebagai *framework*-nya. Dan bahasa *query* yang digunakan baik pada penelitian ini maupun pada penelitian sebelumnya yaitu sama-sama menggunakan bahasa *query* SPARQL.

Untuk hasil dari penelitian sebelumnya yaitu hanya berupa pengelolaan *knowledge* tiket kereta api dalam protégé 3.4 dan pencarian dilakukan secara semantik, namun belum berbasis web. Sedangkan dalam penelitian ini tidak hanya sampai pengelolaan *knowledge* koleksi perpustakaan pada protégé 3.4 saja, namun pencariannya didasarkan pada kriteria yang telah ditentukan dan juga secara semantik.

2.1 Representasi Pengetahuan

Ada beberapa teknik representasi pengetahuan yaitu Logika, Pohon, Naskah, Frame dan Jaringan Semantik. Yang digunakan sebagai acuan dalam pembuatan semantic web adalah menggunakan teknik representasi pengetahuan menggunakan jaringan semantic atau semantic net.

2.2 Jaringan Semantik (Semantic Net)

Semantic Network pertama kali dikembangkan untuk *Artificial Intelligence* (AI) sebagai cara untuk merepresentasikan memori dan pemahaman bahasa manusia. Struktur *semantic net* berupa grafik dengan *node* (simpul) dan *arc* (ruas) yang menghubungkannya. *Nodes* adalah obyek sedangkan *arc* sebagai penghubung (*link*). *Link* digunakan untuk menunjukkan relasi, dan *node* merepresentasikan obyek fisik, konsep atau situasi.

Relasi adalah hal yang sangat penting dalam *semantic net* karena memberikan struktur dasar untuk mengorganisasikan pengetahuan. Tanpa relasi, pengetahuan hanya sekedar koleksi fakta yang tidak saling berkaitan. Dengan relasi, pengetahuan adalah struktur yang kohesif mengenai pengetahuan lain yang dapat disimpulkan dari relasi itu. *Semantic net* kadang disebut juga sebagai *assosiative nets* karena *node*-nya diasosiasikan atau terhubung dengan *node* lainnya.

2.3 Teknologi Web 3.0

Teknologi web 3.0 atau biasa disebut semantic web yaitu web yang memiliki makna, selain membuat mesin mengerti dengan apa yang kita inginkan, web 3.0 sudah memakai *knowledge base* sebagai menyimpan *knowledge* yang nantinya manusia akan mengetahui tentang sesuatu yang dicari tidak hanya berdasarkan data dan informasi tetapi *knowledge* nya juga akan diketahui. Web 3.0 sedang dikembangkan oleh Google yaitu Goolge co-op.

2.4 Semantic Web

Semantic Web merupakan visi masa depan web, dan informasi diberi arti eksplisit, sehingga lebih mudah diproses oleh mesin secara otomatis dan lebih mudah menyatukan informasi yang tersedia di web. *Semantic Web* adalah perluasan dari web yang mendukung database dalam bentuk yang dapat dibaca oleh mesin (Berners-Lee dkk, 2001).

2.5 Perbedaan Sistem Pakar dengan Semantic Web

Semantic Web menggunakan ontologi sebagai domain pengetahuannya dan ontologi mengacu Semantik Net atau pemodelan semantik yang merupakan bagian dari bidang terapan kecerdasan buatan yaitu pemrosesan bahasa alami, pemodelan semantik dan mesin yang dapat belajar. Proses penelusuran dalam sistem pakar menggunakan aturan (*rule based*) sedangkan *semantic web* menggunakan makna yang saling berelasi sehingga menjadi suatu pengetahuan.

2.6 Resource Description Framework (RDF)

RDF adalah bagian utama dari *Semantic Web* yaitu format untuk menyediakan informasi dalam bentuk yang dapat dibaca oleh mesin. Menggunakan URL dan strings teks sebagai istilah, RDF merupakan sebuah model sederhana untuk mendeskripsikan hubungan antar *resources*. Dengan format RDF memungkinkan mengkombinasikan dua dokumen XML menjadi satu dan mendeskripsikan relasi di dalam dokumen tersebut.

2.7 Resources Description Framework Schema (RDFS)

Arti dalam RDF dinyatakan melalui keterangan berupa *schema*. *Schema* dapat dianggap kamus. *Schema* ini mendefinisikan istilah-istilah yang akan digunakan dalam pernyataan RDF dan memberikan arti tertentu untuk istilah tersebut. RDFS menyediakan informasi mengenai definisi dan batasan dari properti yang digunakan. Sebagaimana RDF menggunakan fasilitas *namespace* dari XML maka *namespace* ini juga digunakan untuk meningkatkan penggunaan kata tertentu di dalam RDFS agar kata tersebut terdefinisi sesuai keinginan.

2.8 Ontologi

Ontologi merupakan suatu teori tentang makna dari suatu obyek, properti dari satu objek, serta relasi objek tersebut yang mungkin terjadi pada suatu domain pengetahuan. Ontologi membuka kemungkinan untuk

berpindah dari pandangan berorientasi dokumen, ke arah pengetahuan yang saling berkait, dikombinasikan dan dikelola untuk digunakan dikemudian hari (Wiryana dan Hasibuan, 2002).

2.8.1 Komponen Ontologi

1. *Individuals (instances)*

Instances adalah komponen dasar dari suatu ontologi. *Instances* dari sebuah ontologi bisa berupa objek nyata seperti manusia, hewan, meja, molekul atau bisa juga berupa objek abstrak seperti bilangan dan huruf.

2. *Classes (concepts)*

Suatu kelas menjelaskan konsep-konsep yang ada dalam suatu domain. Biasanya sebuah kelas merupakan sekumpulan dari obyek-obyek. Sebuah kelas juga bisa memiliki subkelas yang akan mempresentasikan konsep yang lebih spesifik daripada superkelasnya.

3. *Attributes*

Obyek–obyek yang berada dalam ontologi bisa dideskripsikan dengan memberi bahan tambahan atribut kedalam obyek tersebut. Setiap atribut memiliki paling tidak sebuah nama dan nilai. Dan digunakan untuk menyimpan informasi yang spesifik tentang obyek yang diberi atribut tersebut.

4. *Relationships*

Suatu *relationships* menjadi penting dalam sebuah ontologi, karena dalam suatu ontologi relasi antara obyek-obyek yang ada harus mendeskripsikan keunggulan dari ontologi yang berasal dari kemampuannya mendeskripsikan relasi ini. Bersama-sama kumpulan dari relasi akan mendeskripsikan atau arti dari sebuah entitas.

2.8.2 Tahap-tahap Pengembangan Ontologi

Tahap-tahap pengembangan ontologi oleh Noy dan McGuinness (2001) dalam makalah "*Ontology Development 101: A Guide to Creating your first Ontology*" terdiri dari beberapa tahap, yaitu:

1. Menentukan domain dan ruang lingkup ontologi .
2. Mempertimbangkan kemungkinan penggunaan ontologi yang sudah ada.
3. Menentukan daftar terminologi-terminologi yang penting dalam ontologi.
4. Mendefinisikan kelas dan hirarkinya.
5. Mendefinisikan property dari kelas (*slot*).
6. Mendefinisikan batasan (*facet*) dari *slot*.
7. Membuat *instance-instance*

2.9 Ontologi Web Language (OWL)

OWL dapat digunakan untuk merepresentasikan arti dari istilah-istilah suatu domain pengetahuan secara eksplisit dan relasi antara istilah tersebut. OWL memiliki fasilitas lebih dibanding XML, RDF, RDFS dalam mengekspresikan arti dan semantik. OWL juga melebihi bahasa-bahasa lain tersebut diatas dalam kemampuan merepresentasikan isi dokumen web yang *machine-interpreted* (McGuinness dkk, 2003).

3. METODE PENELITIAN

3.2 Analisis Kebutuhan Perangkat Lunak

Dalam pencarian koleksi pada perpustakaan FTI UPN "Veteran" Yogyakarta mahasiswa yang ingin mencari koleksi tersebut masih melakukan dengan manual yaitu tidak ada sistem khusus yang dapat digunakan untuk melakukan proses pencarian. Oleh karena itu dibuat aplikasi program yang dapat memudahkan melakukan pencarian koleksi pada perpustakaan FTI UPN "Veteran" Yogyakarta baik bagi mahasiswa maupun petugas perpustakaan.

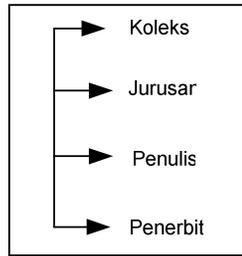
Data perpustakaan FTI UPN "Veteran" Yogyakarta diolah admin untuk dimasukkan sebagai pengetahuan perpustakaan melalui *software protégé*.

3.3 Perancangan

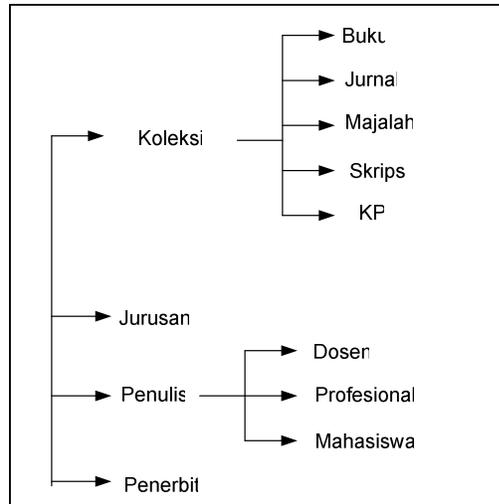
Di dalam tahapan perancangan perangkat lunak ini terdapat tiga tahap perancangan yaitu perancangan *semantic net*, perancangan model ontology dan perancangan penerapan teknologi *semantic web*.

1.3.1 Perancangan Model Ontology

Menentukan ruang lingkup dan domain, Mempertimbangkan penggunaan ontology yang sudah ada, Menentukan daftar terminologi, dan Mendefinisikan class dan hirarkinya



Gambar 3.2 Rancangan Pendefinisian Class



Gambar 3.3 Rancangan Pendefinisian Subclass

Tabel 3.1 Rancangan slot yang terbentuk pada Subclass Buku class Koleksi

Property	Range	Allowed Values	Type
Judul	Single String	Instance	Datatype Property
Jenis	Single String	Buku, Majalah, Jurnal, Skripsi, KP	Datatype Property
Bahasa	Single String	Bahasa Indonesia, Bahasa Inggris	Datatype Property
Warna Cover	Single String	Instance	Datatype Property
Diterbitkan	Multiple Penerbit	Instance	Object Property
Ditulis	Multiple Dosen atau Profesional	Instance	Object Property
Halaman	Single int	Instance	Datatype Property
Jumlah	Single int	Instance	Datatype Property
Tahun Terbit	Single int	Instance	Datatype Property
Jumlah CD	Single int	Instance	Datatype Property
Kode Koleksi	Single String	Instance	Datatype Property
Status	Single String	Bisa dipinjam, Tidak bisa dipinjam	Datatype Property
Koleksi dari	Multiple Jurusan	Instance	Object Property

Tabel 3.2 Rancangan slot yang terbentuk pada Subclass Profesional class Penulis

Property	Range	Allowed Values	Type
Nama	Single String	Instance	Datatype Property
e-mail	Single String	Instance	Datatype Property
Profesi Lain	Single String	Instance	Datatype Property
Menulis	Multiple Buku, Jurnal	Instance	Object Property

Tabel 3.3 Rancangan slot yang terbentuk pada class Penerbit

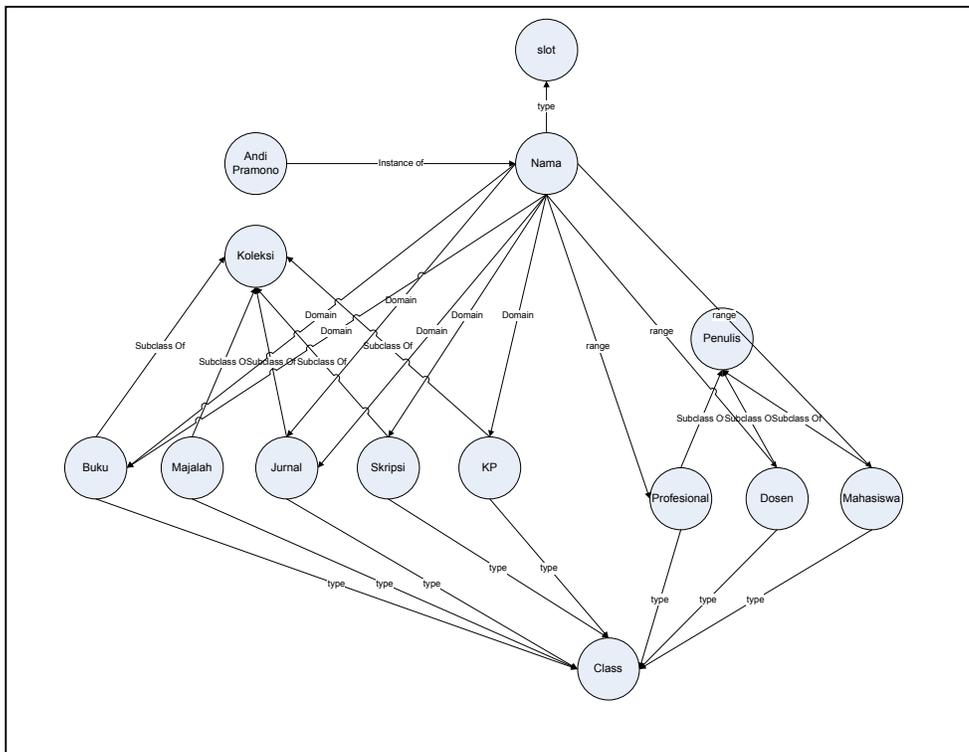
Property	Range	Allowed Values	Type
Nama_penerbit	Single String	Instance	Datatype Property
Menerbitkan	Multiple Buku, Jurnal, Majalah	Instance	Object Property
Alamat	Single String	Instance	Datatype Property
e-mail	Single String	Instance	Datatype Property
No_Telepon	Single String	Instance	Datatype Property
Fax	Single String	Instance	Datatype Property
Website	Single String	Instance	Datatype Property

Tabel 3.4 Rancangan slot yang terbentuk pada class Jurusan

Property	Range	Allowed Values	Type
Nama_jurusan	Single String	Teknik Industri, Teknik Informatika, Teknik Kimia_D3	Datatype Property
Punya_koleksi	Multiple Buku, Jurnal, Majalah, Skripsi, KP	Instance	Object Property

Membuat Instance-Instance

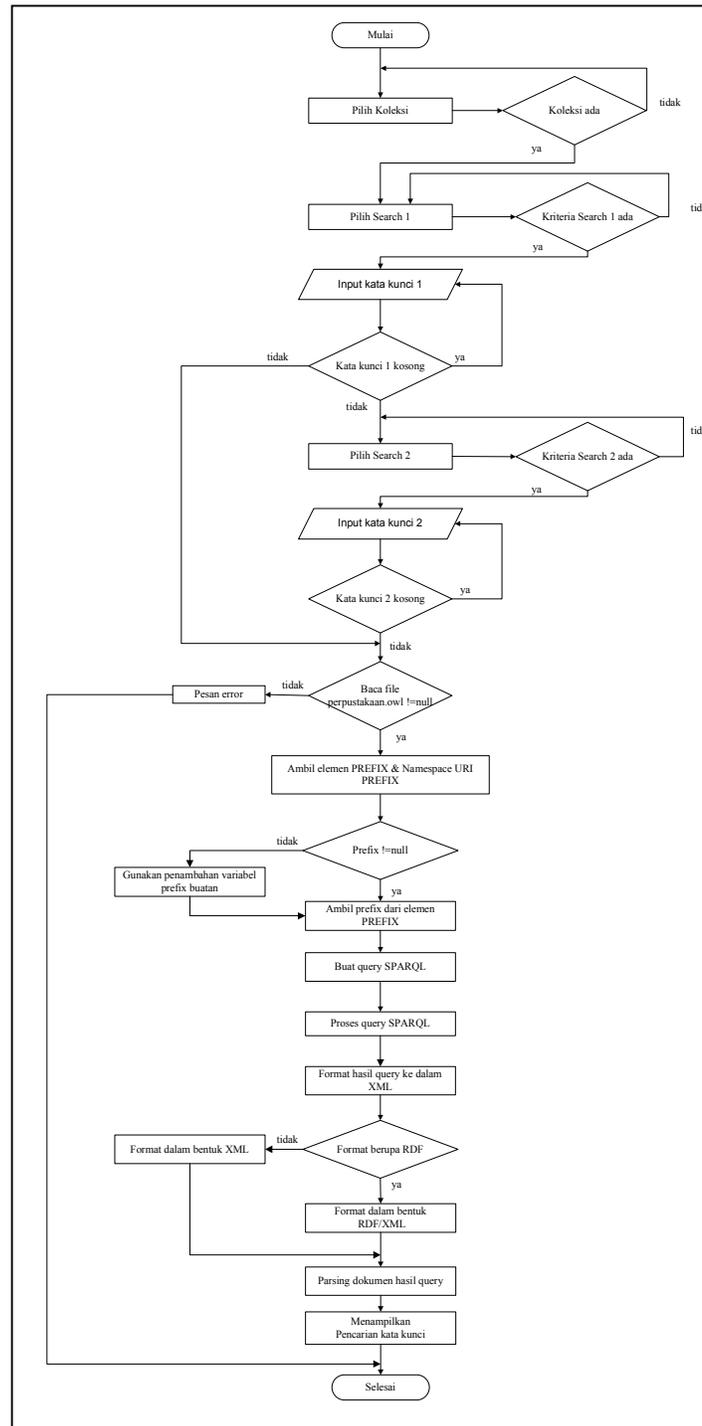
Gambar 3.4 menunjukkan integrasi class Koleksi dan class Penulis. Class Penulis dibagi menjadi subclass Profesional, subclass Dosen, dan subclass Mahasiswa.



Gambar 3.4 Rancangan slot integrasi yang terbentuk pada class Penulis dan class Koleksi

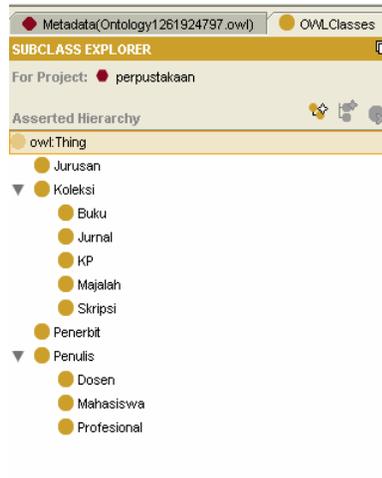
1.3.2 Perancangan Penerapan Teknologi Semantic Web

Flowchart



Gambar 3.8 Flowchart

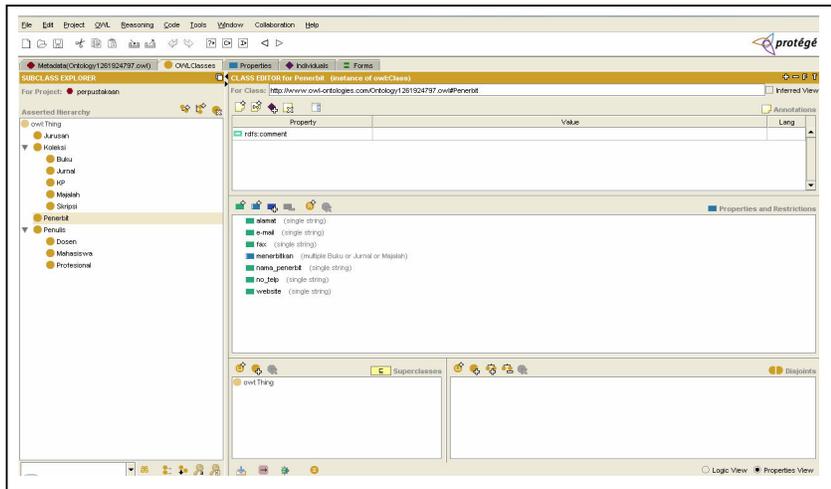
4. HASIL DAN PEMBAHASAN
4.1 Implementasi Class



Gambar 4.1 Pendefinisian Hierarki class dengan Protégé

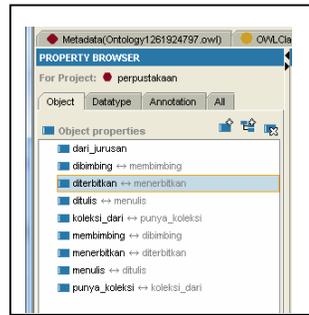
4.2 Implementasi Property pada Class

Pembuatan properti-properti pada class bisa dilakukan melalui tab *Properties*. Dalam protégé 3.4 ini telah dibuat subtab dari tab *Properties* yang terdiri dari tab *Object*, tab *Datatype*, tab *Annotation*, dan tab *All*. Tab *Object* digunakan untuk membuat *object properties*, tab *Datatype* digunakan untuk membuat *datatype properties*, tab *Annotation* digunakan untuk membuat *annotation properties*, sedangkan tab *All* digunakan untuk membuat semua jenis properti yang terintegrasi dalam satu jendela.



Gambar 4.2 Properti-properti pada class Penerbit

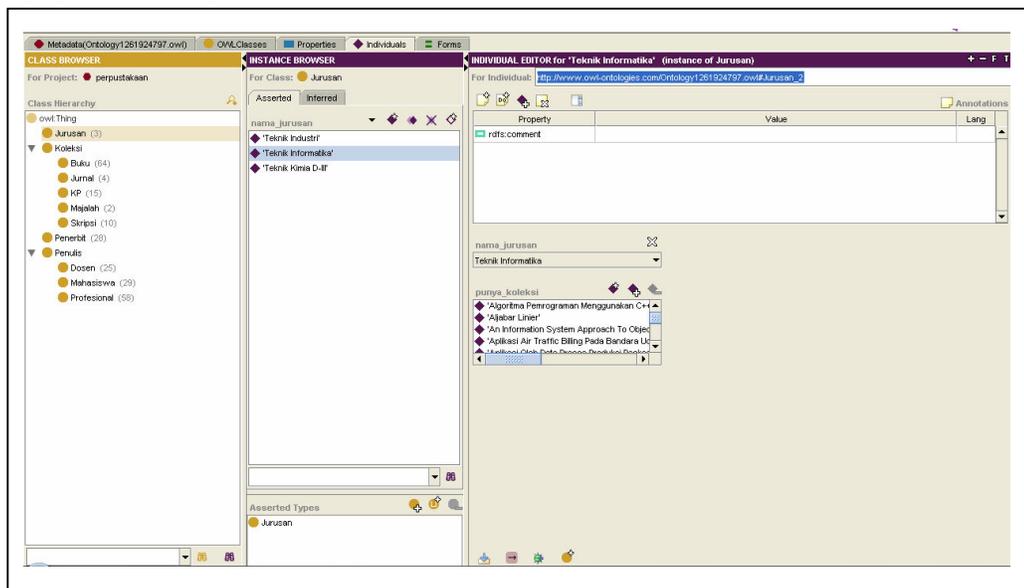
Inverse Property digunakan untuk mendefinisikan properti-properti yang berlawanan. Pada class Penerbit terdapat properti yang menggunakan fungsi *inverse* yaitu, properti menerbitkan yang memiliki *inverse* diterbitkan pada class Buku, Jurnal dan Majalah yang merupakan subclass dari class Koleksi. Dengan adanya fungsi *Inverse Property* maka nilai-nilai dari suatu properti akan secara otomatis terisi ketika *inverse* dari properti tersebut diberi suatu nilai. Sebagai contoh tampak pada Gambar 4.3, properti menerbitkan memiliki *inverse* dengan diterbitkan.



Gambar 4.3 Inverse Properti

4.3. Implementasi Pembuatan Instance

Instance-instance dari class ini merepresentasikan informasi tentang jurusan yang mempunyai koleksi-koleksi yang terdapat pada perpustakaan FTI UPN "Veteran" Yogyakarta. Gambar 4.4 merupakan salah satu contoh pengisian instance pada class Jurusan.



Gambar 4.4 Nilai Properti untuk Instance "Teknik Informatika"

4.4. Implementasi Query

Sebagai contoh query sederhana adalah temukan judul semua koleksi yang menjadi Koleksi dari jurusan "Teknik Informatika". Pada Gambar 4.5 memperlihatkan hasil query sederhana.



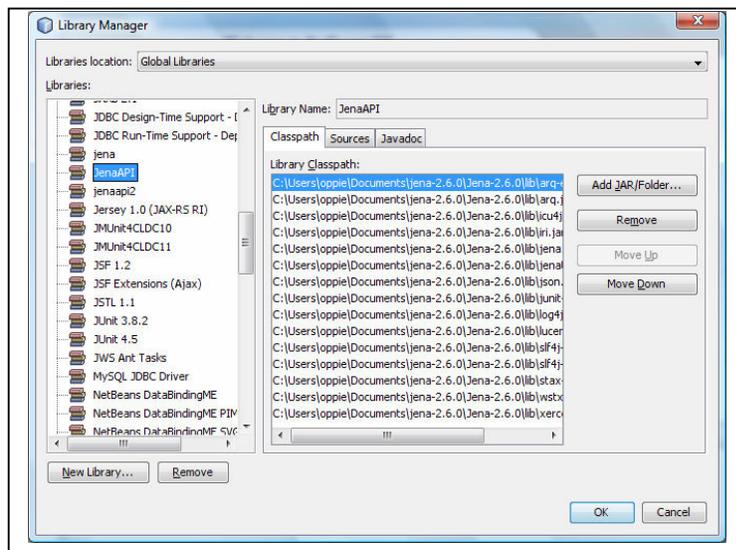
Gambar 4.5 Contoh Query SPARQL Sederhana

```
SELECT ?judulkoleksi
WHERE {
  ?perpus rdf:type ?Koleksi
  .?perpus :judul ?judulkoleksi
  .?perpus :koleksi_dari ?Jurusan
  .?Jurusan :nama_jurusan "Teknik Informatika"}
```

Modul Program 4.1 Contoh Query SPARQL Sederhana

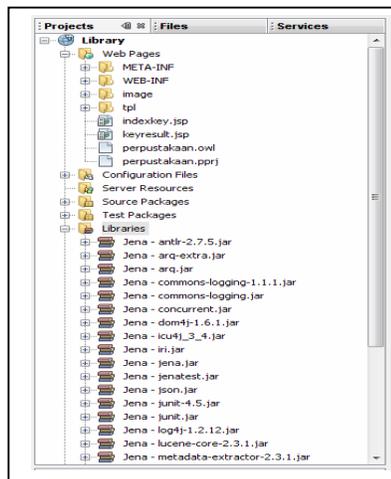
4.2 Implementasi JENA API

Direktori Libraries merupakan tempat meletakkan file-file JAR yang diperlukan oleh aplikasi web. Di direktori ini disimpan file-file JAR standar dari java (JDK) dan *framework* untuk mengembangkan aplikasi berbasis *semantic web* yaitu Jena API. Salah satu langkah untuk mengimport *library* Jena API ke dalam NetBeans bisa dilakukan melalui *Libraries* pada jendela *project* kemudian klik kanan pilih *Add JAR/Folder...* Setelah itu tentukan letak file-file berextension *jar* yang akan ditambahkan dalam *arq-extra.jar*, *dom4j-1.6.1.jar*, *jena.jar* dan sebagainya. Gambar 4.6 memperlihatkan cara menambahkan *library* Jena API ke dalam NetBeans.



Gambar 4.6 Pengimporan Library Jena ke dalam NetBeans

Kemudian akan muncul jendela *Add JAR/Folder* yang digunakan untuk menentukan letak file-file *library* yang akan ditambahkan. Pada Gambar 4.7 memperlihatkan penambahan *library* JenaAPI ke dalam direktori *Libraries*.

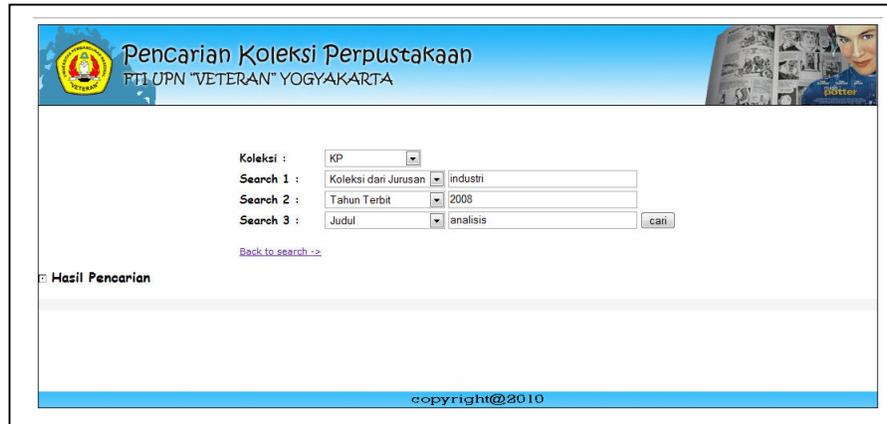


Gambar 4.7 Hasil Penambahan Library Jena API ke dalam Direktori Libraries

4.5 Implementasi Halaman Antarmuka dengan Pencarian dengan Tiga Kriteria

User dapat memasukkan kata kunci pada *dropdown list* dan *text entry field* kriteria yang digunakan sebagai dasar pencarian, yaitu kriteria Koleksi dan Search pertama dan Search kedua yang pilihannya adalah Judul, Penulis, Penerbit, Koleksi dari Jurusan, dan juga Tahun Terbit.

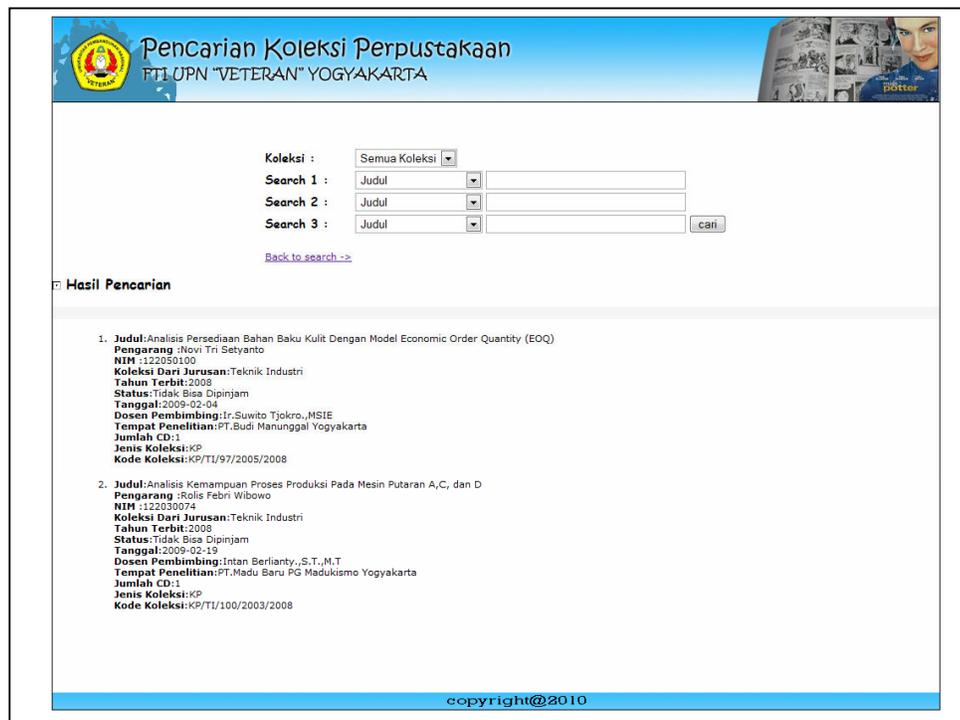
Contoh pencarian menggunakan tiga kriteria pencarian berdasarkan koleksi KP dan berdasarkan kriteria pertama koleksi dari jurusan Teknik Industri, kriteria kedua tahun terbit "2008" dan kriteria ketiga judul "Analisis" seperti pada Gambar 4.12:



The screenshot shows a web interface for searching a library collection. The title is "Pencarian Koleksi Perpustakaan FTI UPN 'VETERAN' YOGYAKARTA". Below the title, there are search criteria: "Koleksi" is set to "KP", "Search 1" is "Koleksi dari Jurusan" with "industri" entered, "Search 2" is "Tahun Terbit" with "2008" entered, and "Search 3" is "Judul" with "analisis" entered. A "cari" button is next to the third search field. Below the search fields, there is a "Back to search ->" link and a section titled "Hasil Pencarian" which is currently empty. The footer contains "copyright@2010".

Gambar 4.12 Halaman Aplikasi Pencarian Koleksi Perpustakaan dengan Menggunakan Tiga Kriteria Pencarian dengan Koleksi KP, Search pertama Koleksi dari Jurusan "Industri", Search kedua Tahun Terbit "2008" dan Search ketiga Judul "Analisis"

Hasil proses pencarian akan ditampilkan seperti terlihat pada Gambar 4.13 :



The screenshot shows the search results page. The search criteria are the same as in Gambar 4.12. Below the search fields, there is a "Back to search ->" link and a section titled "Hasil Pencarian" containing two search results. Each result includes the following information: Judul, Pengarang, NIM, Koleksi Dari Jurusan, Tahun Terbit, Status, Tanggal, Dosen Pembimbing, Tempat Penelitian, Jumlah CD, Jenis Koleksi, and Kode Koleksi. The footer contains "copyright@2010".

- Judul:**Analisis Persediaan Bahan Baku Kulit Dengan Model Economic Order Quantity (EOQ)
Pengarang :Novi Tri Setyanto
NIM :122050100
Koleksi Dari Jurusan:Teknik Industri
Tahun Terbit:2008
Status:Tidak Bisa Dipinjam
Tanggal:2009-02-04
Dosen Pembimbing:Ir.Suwito Tjokro,,MSIE
Tempat Penelitian:PT.Budi Manunggal Yogyakarta
Jumlah CD:1
Jenis Koleksi:KP
Kode Koleksi:KP/TT/97/2005/2008
- Judul:**Analisis Kemampuan Proses Produksi Pada Mesin Putaran A,C, dan D
Pengarang :Rolis Febrin Wibowo
NIM :122030074
Koleksi Dari Jurusan:Teknik Industri
Tahun Terbit:2008
Status:Tidak Bisa Dipinjam
Tanggal:2009-02-19
Dosen Pembimbing:Intan Berlianty,,S.T.,M.T
Tempat Penelitian:PT.Madu Baru PG Madukismo Yogyakarta
Jumlah CD:1
Jenis Koleksi:KP
Kode Koleksi:KP/TT/100/2003/2008

Gambar 4.13 Hasil Pencarian dengan Kriteria Pencarian Koleksi KP, Search pertama Koleksi dari Jurusan "Industri", Search kedua Tahun Terbit "2008" dan Search ketiga Judul "Analisis"

5. KESIMPULAN

Dari pembahasan bab-bab sebelumnya mengenai penerapan teknologi *Semantic Web* pada aplikasi pencarian koleksi perpustakaan maka dapat disimpulkan sebagai berikut :

1. Pengetahuan mengenai informasi pencarian koleksi perpustakaan dapat disimpan dalam model ontologi berbasis OWL dan dibagi ke dalam 4 kelas utama, yaitu *Class Jurusan*, *Class Koleksi*, *Class Penulis*, dan *Class Penerbit* serta dilengkapi dengan pendefinisian properti-properti dari masing-masing kelas.
2. Aplikasi pencarian koleksi perpustakaan dapat dilakukan dengan dua teknik pencarian yaitu pencarian *simple searching* dan *advance searching* dengan memilih beberapa kriteria yang telah ditentukan.
3. Pencarian informasi koleksi perpustakaan dengan memanfaatkan ontologi perpustakaan sebagai basis pengetahuannya mampu membantu *user* untuk menemukan koleksi yang diinginkan yang terdapat pada perpustakaan FTI UPN "Veteran" Yogyakarta khususnya.

6. DAFTAR PUSTAKA

- Azhari dan Wardoyo, 2005, Pendekatan *Model Data Dinamis pada Sistem Basisdata Cerdas*, Prosiding Seminar Nasional: *Revitalisasi Penelitian Bidang Matematika dan Ilmu Pengetahuan Alam*, Dies ke-50, FMIPA UGM, Yogyakarta, 17 September 2005, Hal C1-C8.
- Berners-Lee, T., Connolly, D., Swick, R., 1999, *Web Architecture : Describing and Exchanging Data*, <http://www.w3.org/1999/06/07-Web-Data> (diakses pada tanggal 2 Mei 2009)
- Berners-Lee, T., Hendler, J., Lasilla, O., 2001, *The Semantic Web*, *American Scientific*, http://en.wikipedia.org/wiki/Semantic_Web (diakses pada tanggal 2 Mei 2009)
- Chandarsekaran, B., Josephson, J., 1999, *What are Ontologies, and Why Do Who Need Them?*, *IEEE Intelligent System*, vol 14(1), hal 20-26.
- McGuinness, D., Harmelen, F.V., 2004, *OWL Web Ontology Language Overview*, W3C Recommendation, <http://www.w3.org/TR/2004/REC-owl-feature-20040210> (diakses pada tanggal 3 Mei 2009)