

PENERAPAN WEB SEMANTIK UNTUK APLIKASI PENCARIAN PADA REPOSITORI KOLEKSI PENELITIAN, STUDI KASUS: PROGRAM STUDI SISTEM INFORMASI STMIK MIKROSKIL MEDAN

Gunawan¹, Fandi Halim²

Program Studi Sistem Informasi, STMIK Mikroskil

Jl. Thamrin No. 112, 124, 140 Medan 20212

gunawan@mikroskil.ac.id¹, fandi@mikroskil.ac.id²

Abstrak

Web berbasis semantik adalah suatu jaringan yang mampu memahami tidak hanya makna dari sebuah kata dan konsep, namun juga hubungan logis di antara keduanya, sehingga suatu *web* dapat menghasilkan informasi yang sesuai dan diinginkan oleh pengunjung *website* tersebut. Untuk menunjang kegiatan penelitian pada sebuah perguruan tinggi dapat dikembangkan sebuah repositori koleksi penelitian, dimana repositori koleksi penelitian yang terintegrasi dengan baik bisa membantu peneliti dalam mencari referensi untuk melakukan suatu penelitian. Penelitian ini mengambil studi kasus pada Program Studi S-1 Sistem Informasi STMIK Mikroskil. Saat ini, pengelolaan koleksi penelitian masih sebatas pada pencatatan administratif dan dokumentasi dalam bentuk media CD. Selain itu, dosen dan mahasiswa juga belum difasilitasi dengan media yang dapat dimanfaatkan sebagai sarana untuk mencari dan melakukan perbandingan terhadap koleksi penelitian yang telah pernah dikerjakan sebelumnya ataupun pencarian terhadap koleksi penelitian terbaru. Melihat karakteristik dari permasalahan tersebut, maka repositori koleksi penelitian dapat dikembangkan dengan menggunakan struktur ontologi dan jaringan semantik dari teknologi *web* semantik. Hasil dari penelitian ini berupa repositori koleksi penelitian yang dilengkapi dengan aplikasi pencarian berbasis *web* semantik untuk menunjang program studi, dosen, dan mahasiswa di dalam mengakses koleksi penelitian.

Kata kunci: *web semantik, ontologi, repositori, pencarian*

1. Pendahuluan

Perkembangan teknologi saat ini semakin pesat, dimana hampir semua bidang teknologi sekarang adalah berbasiskan Internet. Salah satu teknologi yang saat ini sangat dirasakan manfaatnya adalah *World Wide Web* atau dikenal dengan nama *web*. *Web* kini menjadi sumber data yang terbesar dan sangat berharga untuk setiap pengguna karena di dalam *web* kumpulan dokumen saling terhubung dan dapat diakses melalui koneksi Internet. Setiap individu bebas menambahkan konten dalam *web*-nya. Semakin banyak informasi yang ditambahkan, semakin besar ukuran *web* dan semakin sulit untuk mencari informasi yang benar-benar diinginkan di *web* dalam waktu yang relatif singkat. Selain itu, faktor keakuratan dari pencarian dan pemilahan suatu informasi juga merupakan suatu hal yang sangat dibutuhkan. Dikarenakan hal tersebut, maka timbul keinginan agar bahasa yang digunakan manusia sehari-hari tidak hanya dapat dimengerti oleh sesama manusia, akan tetapi juga dapat

¹ Gunawan, S.Kom., M.T.I.

² Fandi Halim, S.Kom., M.Sc.

dimengerti oleh mesin (Internet), sehingga segala sesuatu yang disampaikan dapat diterima dengan baik oleh mesin dan informasi yang disajikan pun sesuai dengan yang diinginkan. Oleh karena itu dibutuhkan suatu teknologi untuk mendapatkan informasi yang benar-benar diinginkan tersebut. Di sinilah *web* yang berbasis semantik menjalankan perannya.

Web berbasis semantik adalah suatu jaringan yang mampu memahami tidak hanya makna dari sebuah kata dan konsep, namun juga hubungan logis di antara keduanya, sehingga suatu *web* dapat menghasilkan informasi yang sesuai dan diinginkan oleh pengunjung *website* tersebut. Kelebihan pada *web* semantik adalah untuk menggali sumber ilmu pengetahuan secara *online*, tidak perlu lagi membuka halaman *web* satu per satu, namun dapat memanfaatkan agen untuk menjelajahi ribuan situs-situs *web*. Agen-agen yang cerdas ini akan secara logis hanya memilih hasil-hasil yang relevan sesuai kebutuhan pengguna, kemudian menyajikannya dalam format yang diinginkan. Saat ini dokumen mulai disajikan dalam format *eXtensible Markup Language* (XML). Bahasa lain yang digunakan dalam mendukung visi *web* semantik selain XML adalah *Resource Description Framework* (RDF)/*Ontology Web Language* (OWL). RDF/OWL dengan kemampuan dan fasilitas yang dimilikinya mampu digunakan untuk merepresentasikan makna istilah dalam *web* hingga dapat diproses dalam mesin.

Peran nyata perguruan tinggi dalam pembangunan bangsa terdapat pada konsep “Tri Dharma Perguruan Tinggi”, yang meliputi pendidikan, penelitian, dan pengabdian pada masyarakat. Dari ketiga dharma perguruan tinggi tersebut, fungsi penelitian masih belum terlaksana dengan baik. Untuk menunjang kegiatan penelitian dapat dikembangkan sebuah repositori koleksi penelitian, dimana repositori koleksi penelitian yang terintegrasi dengan baik bisa membantu peneliti dalam mencari referensi untuk melakukan suatu penelitian. Demikian juga halnya pada Program Studi S-1 Sistem Informasi STMIK Mikroskil. Saat ini pengelolaan koleksi penelitian, terutama penulisan ilmiah dan tugas akhir mahasiswa, masih sebatas pada pencatatan administratif di Sistem Informasi Perguruan Tinggi (SIPT) dan dokumentasi dalam bentuk media CD. Hal ini tentunya akan sangat menyulitkan program studi apabila ingin mencari kembali koleksi penelitian untuk kebutuhan tertentu. Selain itu, dosen dan mahasiswa program studi juga belum difasilitasi dengan media yang dapat dimanfaatkan sebagai sarana untuk mencari dan melakukan perbandingan terhadap koleksi penelitian yang telah pernah dikerjakan sebelumnya ataupun pencarian terhadap koleksi penelitian terbaru. Melihat karakteristik dari permasalahan tersebut, maka repositori koleksi penelitian dapat dikembangkan dengan menggunakan struktur ontologi dan jaringan semantik dari teknologi *web* semantik.

Berdasarkan uraian di atas, maka yang menjadi permasalahan adalah belum tersedianya repositori koleksi penelitian yang dilengkapi dengan aplikasi pencarian yang memadai untuk menunjang program studi, dosen, dan mahasiswa di dalam mengakses koleksi penelitian, sehingga terdapat peluang untuk mengembangkan aplikasi tersebut. Teknologi *web* semantik dipilih sebagai salah satu solusinya karena menawarkan kemudahan dan ketepatan dalam proses pencarian informasi. Beberapa hal yang menjadi fokus di dalam penelitian ini adalah sistem yang dibangun hanya untuk fasilitas pencarian koleksi penelitian, pemodelan ontologi dibangun dengan menggunakan *tool* Protégé, yaitu perangkat lunak bantu untuk pengembangan *knowledge base system*, serta bahan pengujian menggunakan sampel tugas akhir sebagai masukan pada sistem yang dibangun.

2. Kajian Pustaka

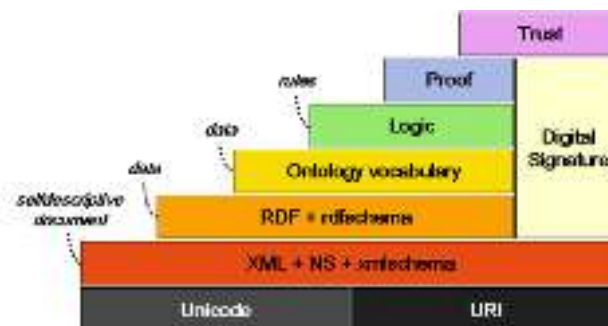
2.1. Web Semantik

Web semantik (semantic web) adalah perkembangan generasi *web* berikutnya atau yang biasa disebut sebagai evolusi dari *World Wide Web (WWW)* yang dicetuskan pada tahun 2002. *Web semantik* pertama kali diperkenalkan oleh Tim Berners-Lee, penemu dari *WWW*, pada tahun 1999. Pengembangan *web semantik* ini didukung oleh *World Wide Web Consortium (W3C)*, yaitu sebuah konsorsium internasional yang dipimpin oleh Tim Berners-Lee dan didirikan pada tahun 1994 dengan misi mengarahkan *WWW* kepada potensi penuhnya dengan mengembangkan protokol-protokol dan pemandu yang menjamin pertumbuhan jangka panjang untuk *web*. *Web semantik* adalah sebuah *web* yang mampu untuk mendeskripsikan sesuatu dalam sebuah cara yang komputer dapat mengerti. *Web semantik* bukan suatu *web* terpisah, tetapi sebuah ekstensi dari yang sudah ada, dimana informasi yang diberikan mempunyai pengertian definisi yang baik, lebih baik untuk memungkinkan komputer dan manusia dalam bekerja sama. Karena sebagian besar perbendaharaan pengetahuan dibangun atas dua pilar utama yaitu semantik dan matematika, maka *web semantik* menyimpan potensi yang sangat besar. Pada *web semantik*, informasi akan diberi *tag* sehingga komputer akan mengerti maknanya. Di dalam membangun sebuah *web semantik*, hal pertama yang perlu dilakukan adalah mendesain struktur semantik dari *web* tersebut [1].

Web semantik didefinisikan sebagai sekumpulan teknologi, dimana memungkinkan komputer memahami arti dari sebuah informasi berdasarkan metadata, yaitu informasi mengenai isi informasi. Dengan adanya metadata, komputer diharapkan mampu mengartikan hasil pemasukan informasi sehingga hasil pencarian menjadi lebih detil dan tepat. W3C mendefinisikan format metadata tersebut adalah *Resource Description Framework (RDF)*. Tiap unit dari *RDF* terdiri dari tiga komposisi, yaitu *subject*, *predicate*, dan *object*. *Subject* dan *predicate* adalah entitas yang ditunjukkan oleh teks, sedangkan *predicate* adalah komposisi yang menerangkan sudut pandang dari *subject* yang dijelaskan *object*. Hal yang paling menarik dari *RDF* yaitu *object* dapat menjadi *subject* yang nantinya diterangkan oleh *object* yang lainnya, sehingga *object* atau masukan dapat diterangkan secara jelas dan detil serta sesuai dengan keinginan pengguna yang memberikan masukan [1].

Untuk mencapai tujuannya dibutuhkan pemberian *meaning* ke dalam masing-masing konten (sebagai atribut) yang akan digunakan oleh teknologi *web semantik*, yaitu ke dalam beberapa *layer* sebagai berikut [1]:

- XML Layer*, untuk merepresentasikan data.
- RDF Layer*, untuk merepresentasikan pengertian dari data.
- Ontology Layer*, untuk merepresentasikan bentuk umum aturan/kesepakatan mengenai pengertian dari data.
- Logic Layer*, untuk menerapkan *intelligent reasoning* dengan data yang bermakna.



Gambar 1 *Layer Web Semantik*

2.2. *Ontology Web Language (OWL)*

OWL adalah bahasa ontologi untuk sebuah *web* semantik yang dikembangkan oleh W3C kelompok kerja *Web Ontology*. Pada mulanya OWL didesain untuk merepresentasikan informasi tentang kategori dari sebuah objek dan bagaimana objek tersebut berhubungan. OWL dapat juga menyediakan informasi tentang objek itu sendiri. Sebagai hasil usaha yang dilakukan oleh kegiatan *Semantic Web* W3C, OWL harus sesuai dengan visi *web* semantik, yaitu bahasa yang dikelompokkan bersama-sama dengan XML dan RDF. OWL yang diharapkan menjadi salah satu bahasa ontologi harus dapat merepresentasikan bagian-bagian yang berguna dalam sebuah ontologi [2].

Dalam usahanya untuk mendukung kemampuan dan skenario yang telah disepakati, OWL menggunakan kemampuan RDF untuk penjabaran statis dan kemampuan struktur kelas serta properti dari skema RDF. OWL dapat mendeklarasikan kelas dan mengorganisasikan kelas tersebut ke dalam hierarki subkelas, sama seperti skema RDF. Kelas OWL dapat dijelaskan sebagai kombinasi logikal (irisian, gabungan, komplemen) dari kelas lainnya, atau sebagai penjelasan satu-persatu dari objek yang dimaksud serta melebihi kemampuan skema RDF. OWL dapat juga mendeklarasikan properti, mengorganisasikan properti tersebut ke dalam hierarki "*subproperty*", dan menyediakan *domain* dan *range* untuk properti tersebut, seperti pada skema RDF. *Domain* dari properti OWL adalah kelas dalam OWL, dan *range* dapat berupa kelas dalam OWL atau tipe data yang dideklarasikan dari luar seperti *string* atau *integer*. OWL dapat menetapkan bahwa properti tersebut adalah transitif, asimetrik, fungsional, atau bertolak belakang dengan properti lainnya [2].

OWL dapat mengekspresikan objek (dapat juga disebut 'individu') mana yang dimiliki oleh kelas yang mana, dan apa nilai properti untuk sebuah individu. Pernyataan yang sama dapat dibuat pada kelas dan properti, pernyataan *disjoint* dapat dibuat pada kelas, serta *equality* dan *inequality* dapat juga disisipkan di antara individu [2].

Kemampuan OWL yang lebih dari RDF adalah kemampuannya untuk menyediakan pembatasan pada bagaimana posisi properti terhadap kelas. OWL dapat mendefinisikan kelas mana yang mempunyai properti terbatas yang membuat semua nilai untuk properti tersebut, sehingga semua nilai untuk properti dalam *instance* harus dimiliki oleh kelas tertentu (atau tipe data) [2].

Beberapa hal yang dapat dilakukan OWL adalah [2]:

- a. Mendeklarasikan kelas, seperti 'negara', 'orang', 'mobil'.
- b. Menyatakan bahwa 'artis' adalah subkelas dari 'orang'.
- c. Menyatakan bahwa 'Indonesia' dan 'Amerika Serikat' adalah anggota kelas 'negara'.
- d. Mendeklarasikan 'bangsa' sebagai properti yang menghubungkan kelas 'orang' (sebagai *domain*) dan kelas 'negara' (sebagai *range*).
- e. Menyatakan bahwa 'umur' adalah properti, dengan 'orang' sebagai *domain* dan *integer* sebagai *range*.
- f. Menyatakan bahwa Johny Depp sebagai anggota dari kelas 'artis'.
- g. Menyatakan bahwa 'negara' dan 'orang' adalah kelas yang *disjoint*.
- h. Menyatakan bahwa 'Kanada' dan 'Jerman' adalah individu yang berbeda.
- i. Mendeklarasikan 'memiliki warga' sebagai kebalikan dari properti 'bangsa'.
- j. Menyatakan bahwa kelas 'tidak ada negara' dibuat untuk semua anggota dari kelas 'orang' yang tidak memiliki nilai untuk properti 'bangsa'.

Elemen-elemen dalam OWL [2]:

- a. *Class*

OWL mendefinisikan *root* dari semua yang ada dengan `owl:Thing`. Jadi semua kelas yang dibuat secara implisit merupakan subkelas `owl:Thing`. Pembuatan kelas menggunakan `owl:Class` dan menyatakan subkelas dengan `rdfs:subClassOf`.

b. *Individual*

Individual atau disebut juga *instance* adalah anggota (*member*) dari kelas. *Instance* ini dapat dipandang sebagai objek yang ada pada *domain* yang dibahas. Sama seperti `owl:Class` yang menjadi *meta level* untuk kelas, begitu pula kelas yang telah didefinisikan menjadi *meta level* untuk *instance*.

c. *Property*

Property merupakan *binary relation*. Ada dua jenis *property* pada OWL, yaitu *ObjectProperty* (relasi antara *instance* dari dua kelas) dan *DatatypeProperty* (relasi antara *instance* dengan RDF literal dan tipe data XML *Schema*). Sama halnya seperti kelas yang dapat dinyatakan secara hierarki, begitu pula *property* dapat dinyatakan sebagai *subPropertyOf* dengan `rdfs:subPropertyOf`. Untuk memberikan batasan pada suatu *property* dapat digunakan `rdfs:domain` dan `rdfs:range` yang disebut juga sebagai *global restriction* karena berlaku untuk umum dan tidak terbatas pada kelas tertentu.

Ada 2 (dua) hal terkait dengan *property*, yaitu:

- a. *Characteristic*, memberikan tambahan keterangan untuk *property*, yaitu *inverseOf*, *TransitiveProperty*, *SymmetricProperty*, *FunctionProperty*, dan *InverseFunctionalProperty*.
- b. *Restriction*, disebut juga sebagai *local restriction* karena memberikan batasan pada definisi suatu kelas, seperti pada contoh yang diberikan sebelumnya tentang kardinalitas dengan *restriction* dan *onProperty*. Ada tiga macam *restriction*, yaitu *quantifier*, *cardinality*, dan *hasValue*. Untuk menentukan *quantifier* digunakan *allValuesFrom* dan *someValuesFrom*.

2.3. Resource Description Framework (RDF)

Ketika mem-publish *linked data* di *web*, informasi tentang sumber daya direpresentasikan dengan menggunakan RDF. RDF menyediakan model data yang sangat sederhana di satu sisi, tetapi sangat disesuaikan terhadap arsitektur *web* di sisi lainnya. Dalam RDF, sebuah deskripsi dari sumber direpresentasikan sebagai sejumlah *triple*. Tiga bagian dari setiap *triple* disebut *subject*, *predicate*, dan *object*. Sebuah contoh struktur dasar *triple* dari kalimat sederhana adalah sebagai berikut [3]:

Chris	has the email address	chris@bizer.de
(subject)	(predicate)	(object)

Subject dari *triple* adalah URI yang mendefinisikan sumber. *Object* dapat berupa nilai *literal* sederhana, seperti *string*, numerik, atau tanggal, atau URI dari sumber daya lainnya yang berkaitan dengan *subject*. *Predicate* mengindikasikan hubungan di antara *subject* dan *predicate*, misalnya ini adalah nama atau tanggal lahir. *Predicate* juga berupa URI. URI *predicate* didapatkan dari kamus data (*vocabularies*), yaitu sejumlah URI yang dapat digunakan untuk merepresentasikan informasi tentang *domain* tertentu [3].

RDF adalah ide dasar dalam mendefinisikan sesuatu menggunakan URI dan mendeskripsikan sumber daya dalam istilah-istilah dari properti sederhana dan nilai properti. Ini memungkinkan RDF untuk merepresentasikan pernyataan-pernyataan sederhana tentang sumber daya sebagai *graph* dari *node* dan *arc* yang merepresentasikan sumber daya, dan properti-propertinya serta nilainya. Pernyataan berikut: "seseorang yang diidentifikasi dengan `http://www.w3.org/People/EM/contact#me`, yang bernama Eric Miller, memiliki alamat email `em@w3.org` dan memiliki gelar Dr" dapat direpresentasikan dalam RDF *graph* sebagai berikut [4].



Gambar 2 RDF *Graph* Yang Mendeskripsikan Eric Miller

RDF juga menyediakan sebuah sintaks berbasis XML (disebut juga RDF/XML) untuk menyimpan data dan pertukaran data [4].

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdfsyntax-ns#"
xmlns:contact="http://www.w3.org/2000/10/swap/pim/contact#">
<contact:Person rdf:about="http://www.w3.org/People/EM/contact#me">
<contact:fullName>Eric Miller</contact:fullName>
<contact:mailbox rdf:resource="mailto:em@w3.org"/>
<contact:personalTitle>Dr.</contact:personalTitle>
</contact:Person>
</rdf:RDF>
```

Gambar 3 Contoh Sintaks RDF Yang Mendeskripsikan Eric Miller

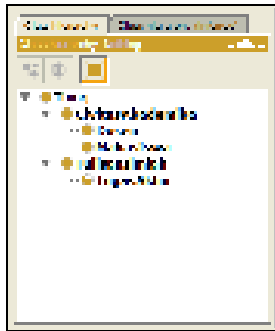
3. Metode Penelitian

Penelitian ini dibagi dalam tiga tahapan utama. Tahapan pertama dari penelitian adalah sebagai tahapan pra-penelitian, yaitu tahapan untuk persiapan penelitian. Pada tahap ini diawali dengan pemilihan topik dan dilanjutkan dengan penyampaian proposal.

Tahapan kedua adalah sebagai tahapan penelitian. Beberapa langkah yang ditempuh antara lain melakukan studi literatur, pengumpulan data, dan desain penelitian. Pada studi literatur, dilakukan pencarian bahan yang berhubungan dengan topik penelitian melalui buku, jurnal, dan *website*. Pada pengumpulan data dan desain penelitian dilakukan beberapa hal, yaitu memodelkan ontologi aplikasi (mendefinisikan taksonomi untuk aplikasi *web* semantik yang akan dibangun), memodelkan ontologi konten, dan memodelkan sampel konten. Setelah itu, ontologi aplikasi dan ontologi konten digabungkan sehingga data dapat terintegrasi secara semantik melalui perancangan RDF. Selanjutnya dirancang *query* yang berfungsi untuk mengambil pengetahuan yang ada pada RDF tersebut dengan menggunakan SPARQL. Langkah berikutnya adalah mengembangkan aplikasi pencarian dengan menggunakan bahasa pemrograman *web* PHP, namun sebelumnya dilakukan konfigurasi ARC2 (*toolkit web* semantik yang mendukung bahasa pemrograman *web* PHP, <https://github.com/semsol/arc2/>). Setelah itu dilakukan pengujian terhadap aplikasi yang telah dikembangkan.

Tahapan ketiga adalah tahapan pasca penelitian. Pada tahap ini diambil kesimpulan dari pengujian-pengujian yang sudah dilakukan dan pemberian saran.

4. Hasil dan Pembahasan
4.1. Perancangan Ontologi



Gambar 4 Susunan Class pada Protégé

Untuk mengembangkan aplikasi pencarian pada repositori koleksi penelitian berbasis *web* semantik diperlukan sebuah ontologi yang merupakan teknologi *web* semantik yang digunakan untuk merepresentasikan pengetahuan apa saja yang diperlukan dalam membangun aplikasi pencarian tersebut. Dalam penelitian ini, ontologi diberi nama “repositori”.

Adapun struktur ontologinya adalah sebagai berikut:

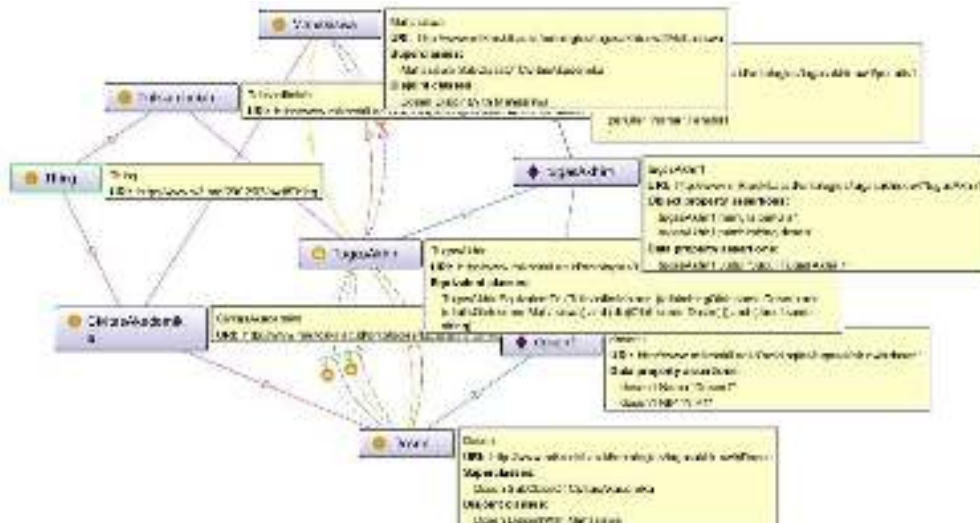
- a. Class CivitasAkademika, terdiri dari subclass:
 1. Dosen, berisi *datatype* nip dan nama.
 2. Mahasiswa, berisi *datatype* nim dan nama.
- b. Class TulisanIlmiah, terdiri dari subclass TugasAkhir, berisi *datatype* ta dan judul.

Dari kelas dan *datatype* yang sudah didefinisikan, kemudian diperlukan suatu relasi yang menghubungkannya. Class bertindak sebagai subjek ontologi dan *datatype* merupakan objek dalam ontologi. Predikat dalam ontologi didefinisikan sebagai penghubung antara subjek dan objek agar memiliki arti kesatuan. Adapun predikat dari ontologi merupakan objek ontologi, dimana bentuknya seperti tabel berikut ini.

Tabel 1 Domain Range “repositori” Ontology

Domain	Object Ontology	Range
TugasAkhir	ditulisOleh	Mahasiswa
Mahasiswa	menulis	TugasAkhir
TugasAkhir	dibimbingOleh	Dosen
Dosen	membimbing	Mahasiswa
TugasAkhir	diujiOleh	Dosen
Dosen	menguji	Mahasiswa

Skema dari ontologi repositori yang dirancang dengan Protégé ditunjukkan pada gambar berikut ini.



Gambar 5 Skema Ontologi “repositori”

4.2. Perancangan RDF

Setelah ontologi dirancang, selanjutnya adalah memasukkannya ke dalam RDF *data store*, yaitu MySQL, dimana akan diperoleh bentuk RDF *database*, *prefix*, *namespace*, dan dapat dilakukan simulasi *query* SPARQL di dalamnya. Tersedia 2 (dua) cara untuk memasukkan data OWL/RDF ke dalam data MySQL OWL/RDF, yaitu melalui *command line* atau melalui kode aplikasi PHP. Pada penelitian ini digunakan kode aplikasi PHP sebagai berikut: `$store->query('LOAD <http://localhost:8080/repositori.owl>');`;

4.3. Perancangan Query

Rancangan *query* terdiri dari 3 (tiga) jenis, yaitu perancangan *query* untuk pencarian berdasarkan kata kunci, *query* untuk pencarian berdasarkan pencarian *simple*, dan *query* untuk pencarian *advanced*.

4.4. Perancangan Aplikasi Pencarian

Aplikasi yang dirancang memiliki 3 (tiga) menu, yaitu Pencarian Kata Kunci, Pencarian *Simple*, dan Pencarian *Advanced*. Gambaran umum dari rancangan aplikasi pencarian ditunjukkan pada gambar-gambar berikut ini.

The figure shows three screenshots of a search application interface, labeled (a), (b), and (c). Each screenshot has a header with the text "Pencarian Kata Kunci || Pencarian Simple || Pencarian Advanced".

(a) Pencarian Kata Kunci: This interface features a "Kata Kunci" label followed by a text input field containing "Masukkan kata kunci" and a "Cari" button. Below the input field are two horizontal lines labeled "Hasil Pencarian 1" and "Hasil Pencarian 2".

(b) Pencarian Simple: This interface includes a dropdown menu on the left and a text input field containing "Masukkan kata kunci" with a "Cari" button. It also has two horizontal lines for search results labeled "Hasil Pencarian 1" and "Hasil Pencarian 2".

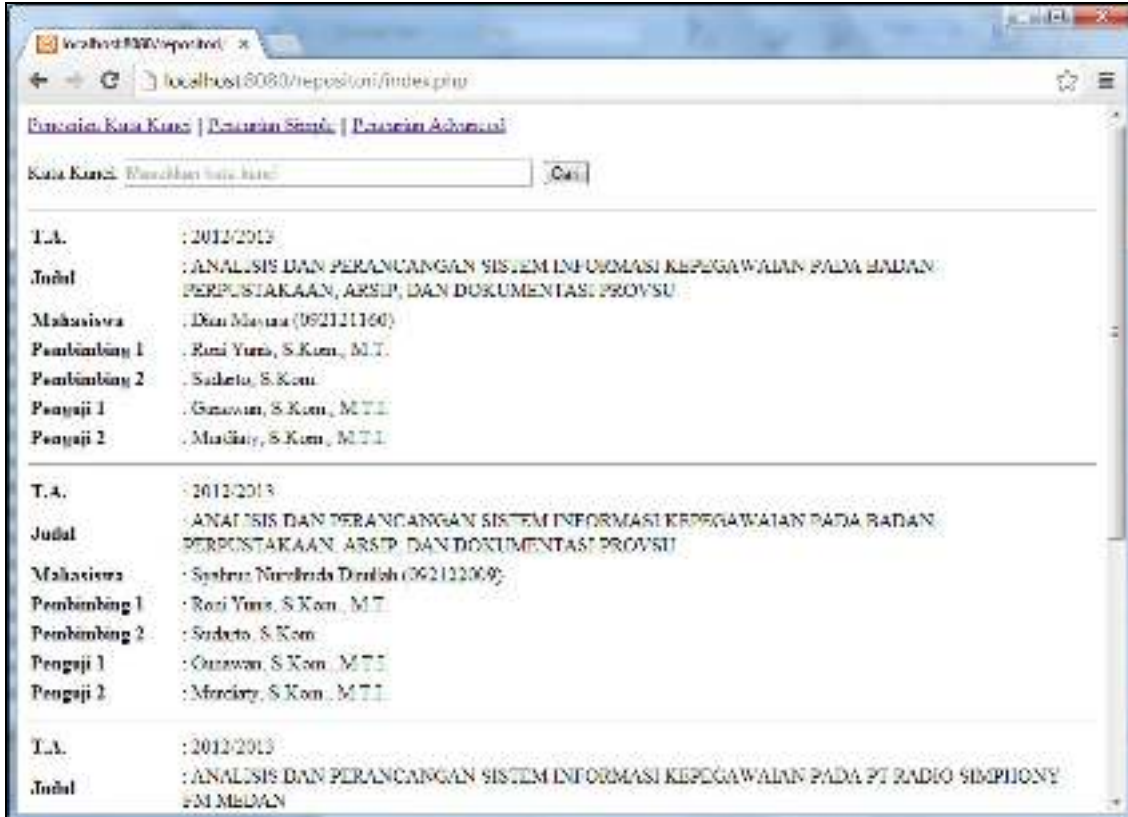
(c) Pencarian Advanced: This interface has three checkboxes with corresponding text input fields: "Judul Tugas Akhir" (Masukkan judul tugas akhir), "Pembimbing" (Masukkan pembimbing tugas akhir), and "Penguji" (Masukkan penguji tugas akhir). A "Cari" button is located below these fields. At the bottom, there are two horizontal lines for search results labeled "Hasil Pencarian 1" and "Hasil Pencarian 2".

Gambar 6 Rancangan Halaman Pencarian

4.5. Implementasi

Untuk melakukan pencarian, dapat dilakukan dengan memilih salah satu menu yang terdapat di bagian atas halaman, yaitu Pencarian Kata Kunci, Pencarian *Simple*, dan Pencarian *Advanced*. Pencarian Kata Kunci adalah fasilitas pencarian dimana pengguna dapat memasukkan beberapa kata kunci (*keyword*) yang berhubungan dengan koleksi penelitian. Pencarian *Simple* adalah fasilitas pencarian berdasarkan kategori pencarian yang telah disediakan, sedangkan pencarian *advanced* adalah fasilitas pencarian dengan menggabungkan beberapa kategori pencarian. Pada contoh yang ditunjukkan berikut ini, untuk Pencarian Kata

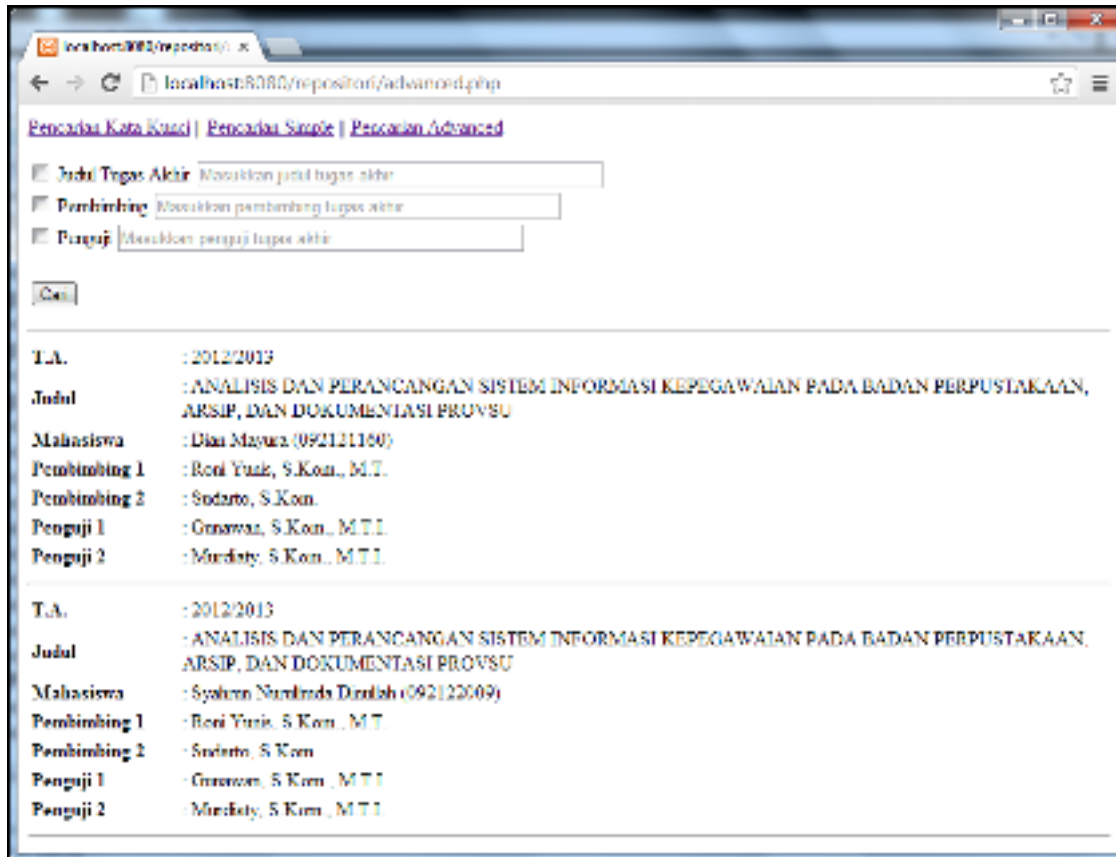
Kunci menggunakan kata kunci “kepegawaian”, untuk Pencarian *Simple* berdasarkan kategori “Mahasiswa” dengan kata kunci “syahrin”, sedangkan untuk Pencarian *Advanced* mengkombinasikan kategori “Judul Tugas Akhir” (kata kunci “kepegawaian”) dan “Pembimbing” (kata kunci “Roni Yunis”).



Gambar 7 Tampilan Halaman Pencarian Kata Kunci



Gambar 8 Tampilan Halaman Pencarian *Simple*

Gambar 9 Tampilan Halaman Pencarian *Advanced*

5. Kesimpulan

Hasil penelitian menunjukkan bahwa teknologi *web* semantik dapat diterapkan sebagai salah satu teknologi untuk aplikasi pencarian, namun masih banyak lagi yang dapat dikembangkan untuk menjadi sebuah aplikasi yang baik. Untuk pengembangan ke depannya dapat dilakukan pemodelan pencarian dengan penambahan *vocabulary* kata kunci, sehingga pencarian dapat dilakukan dengan padanan kata dari suatu kata kunci. Penambahan data yang lebih banyak dan beragam dapat dilakukan untuk mengetahui hasil kecenderungan dari pengukurannya lebih lanjut, sehingga dapat diketahui perbedaan di antara hasil penelitian ini.

Referensi

- [1] W3C, *Semantic Web*, <http://www.w3.org/2001/12/semweb-fin/w3csw/>.
- [2] W3C, *OWL 2 Web Ontology Language Primer*, <http://www.w3.org/TR/2009/REC-owl2-primer-20091027/>.
- [3] Bizer, C., R. Cyganiak, and T. Heath, *How to Publish Linked Data on the Web*, <http://www4.wiwiw.fu-berlin.de/bizer/pub/LinkedDataTutorial/#datamodel>.
- [4] W3C, *RDF Primer*, <http://www.w3.org/TR/2004/REC-rdf-primer-20040210/>.
- [5] W3C, *SPARQL Query Language for RDF*, <http://www.w3.org/TR/rdf-sparql-query/>.
- [6] Protégé, *Open source ontology editor and knowledge-base framework*, <http://protege.stanford.edu/>.