Implementation of Web Ontology and Semantic Application for Electronic Journal Citation System

Riri Fitri Sari

Department of Electrical Engineering, Faculty of Engineering, University of Indonesia
Kampus Baru UI Depok, 16424, Indonesia
Email : riri@ui.ac.id

Nuriana Ayuningtyas

Department of Electrical Engineering, Faculty of Engineering, University of Indonesia
Kampus Baru UI Depok, 16424, Indonesia
Email : nuriana.ayuningtyas@ui.ac.id

Abstract - **This paper reviews web ontology and semantic application to create Indonesian electronic journal citation system. Ontology describes basic the concepts in a domain and defines the components relationship. We have created the basic building blocks of ontology design which include class, slot and instance. The aim of the semantic application built in this work is to classify information from an article in electronic journal and relate it with articles cited in it. The generated web allows user to search the relation among articles in electronic journal cited by particular article.**

The system has been implemented by ontology tool i.e, Protégé and the Mashup has been developed by PHP and MySQL. To update the application, Protege should be installed and used by domain expert or librarian. The result of the ontology model is in XML file. Librarian can submit the result through the web in the form of a system to upload and parse XML file into the database. User could find the citation for an article they searched for through the web based system.

Testing has been performed by ten users to be able to find out the effectiveness of the system for the domain expert or librarian and for the web users. The results show that Protégé is an effective ontology editor to construct domain models for web ontology and the user interface generated in this system works well.

Index Terms - **Ontology, semantic web, Protégé, citation, XML.**

I. INTRODUCTION

Web is a space where information resources are identified by Uniform Resource Identifier (URI). Presently, website has been used by modern society to conduct transactions, as well as to spread and search for information. Search engines such as Google have become the major tool for information searching in the world due to the large number of indexed information. However, although the search engine can provide the information, often the search result is not accurate. The reason behind that is that the Internet does not have mechanism to organized information integration. In other word, World Wide Web is like an incomplete map of geography. To map the region more accurately, the software requires a description of the content and the ability to access web resources that can be read by a machine. Ontology is an ideal concept to describe the structure and semantics of document types. It is expected that in the future information can be given explicitly to create the engine for integration. Computer should understand and be able to process information automatically to integrate the existing information on the web. In addition to that computers are expected to make the process of reasoning, as humans do.

Website's search engine is often used by modern society to search for literatures in the making of a scientific paper. Literature can be an article from journal that later will become a reference for other scientific paper for the fact that the paper contains relevant citation information. Citation to a paper shows that the paper has been acknowledged for its contribution. The relevance of the scientific article to be used as reference for other articles can be done using the ontology which will be described by the structure and the semantic of the e-journal domain.

Currently there are some engines which can provide citation information in the web, i.e. CiteSeer and GoogleScholar. CiteSeer is a web site providing reference data in computer science area taking into account that this is the area in which timely publication can be gathered. Citeseer performs an automated search of the Internet to harvest research papers that have been posted on universities home pages and research centers with information of the authors, titles and citation. The data is then analyzed, enabling the site to rank papers according to how many citations they have received.

Google Scholar is the most widely used freely available search engine for scientific and academic literature. The citations generally used to calculate citation statistics. Google has agreements with publishers to allow Google Scholar to access their private repositories and use robot crawler to search for content. The robot crawler finds and retrieves page on the web and gives the indexer the full

text of the pages it finds to be stored in Google's index database. This index is sorted alphabetically by search term for rapid access to documents that contain user query terms.

Compared to CiteSeer and Google Scholar, our system is a web ontology application which uses the librarian's knowledge in managing data and retrieved the relationship of information. Our system is not only used for computer science paper, but for all available e-journal with fulltext which include the references. We implemented the ontology approach to our problem. An administrator must be there to maintain the system.

This paper presents the literature review on ontology, the reasons for using the ontology, ontology components and their development. Subsequently we will show the design, implementation, and evaluation of our system.

II.   LITERATURE REVIEW

The contribution of this work is to design ontology modeling and transform them into portal viewer in which users can interact with the data provided by ontology. Citation by ontology had not been created before hand. We model an ontology as a base for database citation literature. Protégé is used to introduce the ontology tools and semantic data. The interface is easy enough for Librarian to handle. However the protégé software must be installed locally in the Librarian computer, and then the result can be uploaded to the web based system. Web based format of Protégé such as POWL can be used. However in this first implementation, we decided to use the locally installed protégé software for the benefit of miscellaneous output format which can be used for further work.
The following are some reviews of the ontology and its development theory.

A. The Ontology

In artificial intelligent, there are several understanding of ontology. Ontology is a term borrowed from philosophy that refers to describe the type of entities in the world and how they relate to each other. Barnaras on Kactus project [3] provides the definition of ontology as "An explanation of the concept of representation of knowledge in the knowledge base". Sensus project [2] also gives the definition "ontology is a hierarchical structure of the term to describe a domain that can be used as a foundation for a knowledge base". Other presents ontology as a formal description that describes a concept in a particular domain (**classes**, called **concepts**), properties of each concept that describes the various features and attributes of the concepts (**slots**, called role or **properties**) and restriction on slots (**facets**, called **role restriction**). An ontology together with a set of instances (object expressed in a domain) of the class form a knowledge base. Other opinion stated that the ontology formed by 4 tuples (**C, R, I, A**), in which **C** is a concept, **R** is a relation, **I** is the instance and **A** is the axiom. Axiom is used to provide information about the classes and properties, for example restrictions on the properties.

B. Reasons for Using the Ontology

There are several reasons for using the ontology [4]:

- Explain a domain explicitly. Provide a hierarchy structure of the term to describe a domain and how they relate.
- Share the understanding of the structured information. For example some websites those have information to provide medical services or medical e-commerce. If the website is used and published together with the same ontology, the software can extract and collect information from different site. The software can use information to answer the user's request or input data for other applications.
- Re-use of domain knowledge. If we want to build a broad ontology, we can integrate some existing ontology.

C. Ontology Components

Ontology has several components to explain itself, i.e.:

- **Instance** or **individual** is used to represent element in a particular domain.
- **Class** is a center of ontology. Class explains a concept in a particular domain which contains some instance or individual. Class also known as **concept, object** and **categories**. A class has *subclass that means* more specific concept.
- *Properties* or *slot* consist of two types, the object properties and data type properties. Object properties link between instance while data type properties will connect instance with data type value as text sting and number. Figure 1 depicts the entire ontology components in our system.
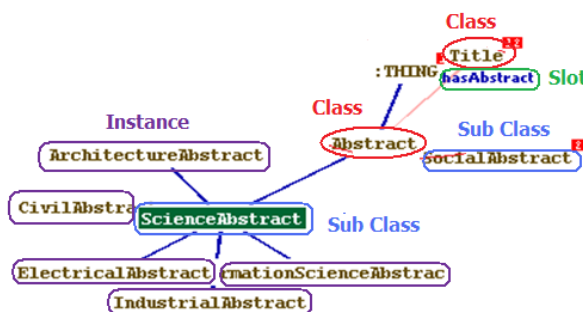

Figure 1. Ontology Components

D. Ontology Development

There are several stages in the development of ontology [6]:

- Determine the domain and range of ontology.
  The development of ontology starts with defining the domain and answers question such as what domain will the ontology cover? why ontology is used? Which kinds of questions is the ontology supposed to answer? One way in determining the ontology range is to create a list of questions which can be answered by knowledge base or commonly called competency questions [6].
- Consider the use of the existing ontology.

Ontology that already exists can be extended for domain and task will be made. The use of ontology is a requirement when the system will interact with other applications that have been developed on an ontology or a controlled vocabulary.

- Define the important terms in the ontology.

  Write a list of terms that will be explained to users such as what terms will be discussed? What will be the answer about the terms? What property owned by any of these terms?

  For example in the wine ontology, terms that are important include wine, place where wine was made, wine color, shape, taste and sugar contents. Basically, this is needed to obtain a comprehensive list of terms without worrying the overlap between the term and concept of the property, or concept includes slot or class.

- Defining the class ontology and composing the class in taxonomy hierarchy (sub class-super class).

  There are several approaches in the development of class hierarchies [5]:
  - **Top-down** development starts with defining the concept in the public domain followed by more specific concept.
  - **Bottom-up** development starts with defining the most specific class and then grouped into classes with more general concept.
  - **Combination** development is a combination of top-down and bottom-up approach. Defining the concept of the prominent first and then generalized the specific concept.

- Defining slots and properties and describes the slot's value.

- Defining facets in the slots.

  Slots have different facets describing the value type; the value may be cardinality (number of values)

  The following is some facet of common use:
  - Slot cardinality

    Slot cardinality defines the number of slot's value. A system consist of single cardinality (have one value) and multiple cardinality (have some value).
  - Slot value type

    A type of facet describes the type of values that can fill the slot. Here is some of value type commonly used:
    1. *String*, the most simple type value for the slot.
    2. *Number,* more specific value type such as *Float and Integer* that is used to describe slot with numeric value.
    3. *Boolean*

       Booleans slot has yes-no flag.
    4. *Enumerated*

       Enumerated slot list of specific value for the slot.
    5. *Instance*

       Instance defined some relationship between individuals.

- Creating *instance*s

The last step is to create individual or instance on the class in a hierarchy. Defining the instance should follow the following method.
- Elect the class class.
- Create instance in the class. Knowledge base may be made by defining the instance of the class that filled the slot on the specific values and additional slot restriction (facets).
- Fill the value of slot.

III. DESIGNING WEB ONTOLOGY AND SEMANTIC APPLICATION

A. System Requirement.

Web ontology and semantic application has the requirement for a reference in the design. The system will be built in the server and information can be accessed by users through the web browser. Figure 2 depicted the general architecture of the system.
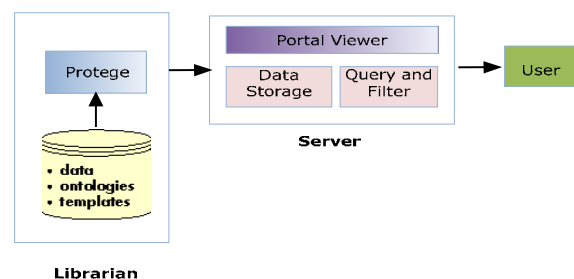

Figure 2. Architecture System

B. System Modeling

As part of the system requirements and design activities, the mashup created in this work consists of several components that are associated, namely:

1. Component to design ontology model. This component is used to design ontology model based on domain expert. In modeling the ontology we used an open source tool known as a Protégé. Result from the design model is structured in the taxonomy hierarchy. The process of designing the model can only be done by a domain expert (in our case a librarian) in other words people who really control a domain. The role of domain expert is to create a taxonomy hierarchy (super class-subclass), property and instance by using the Protégé. Once a complete ontology model is created, the ontology models are stored in XML file.

2. Domain expert interface component.

   This component is used to upload and parse the XML file into MySQL database by a domain expert.

3. Portal component.

   This portal is created with the goal as a mediator for the user to set up a query into the system to get the information they searched for. Figure 3 and 4 describe the schema system in a whole.

   In this work we decided to transform the RDF XML files from Protégé directly into the database. Previously we decided to use RDF [4, 5, 6, 7, 8], however, it was difficult to classify the sentence format and parse the

query due to the fact that the format is in subject, predicate, and object. In the XML format, we could decide our own format, and we could parsed the title and the author information. XML format makes it clearer and easy to create the query. We did not use existing abstraction framework like Sesame or Jena which allow manipulating RDF data. Jena is based on Java, whereas we decided to use XML for the familiarity. It is simpler to use XML format generated from Protégé to create a query.
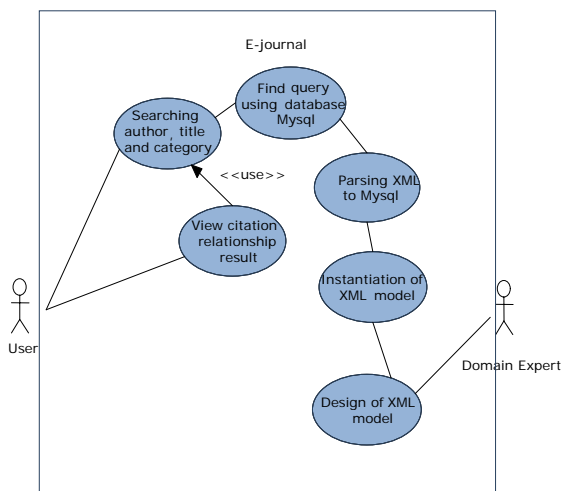

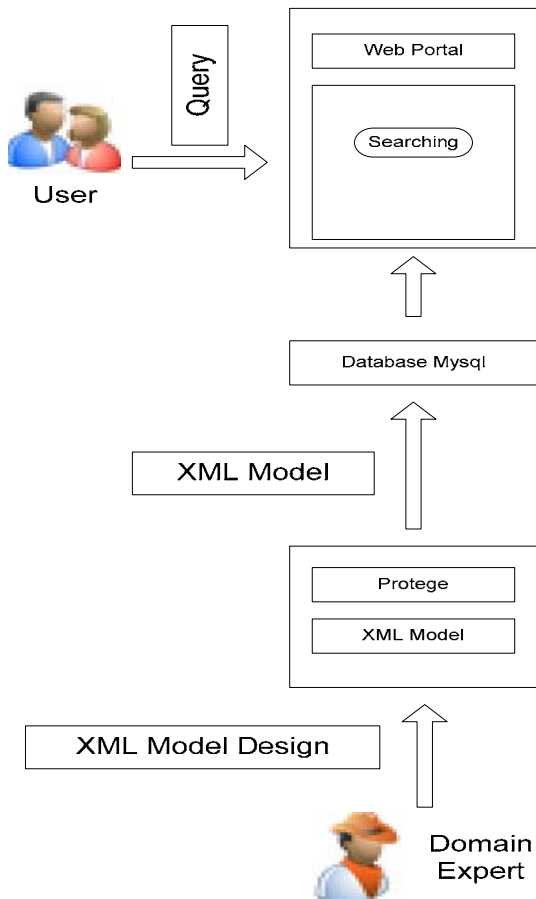Figure 3. Use Case Diagram E-Journal

Figure 4. Schema of Web Ontology and Semantic Application

C. Ontology Design

Ontology design provides information of the stages in the development of ontology and explains the components needed in a representation of information. Stages in the development of ontology are as follows:

1. Determining the domain.
   Domain ontology is Indonesian electronic journal.
2. Defining ontology class and arrange them in taxonomy hierarchy (subclass-superclass) using the process of top-down starts with defining the common concept followed by more specific concept.
3. Defining the slot or property.
4. Defining facets in the slot or axiom in properties.
   Properties have specific domain and range. Properties connect instance in domain and instance in range.
5. Creating instance,
6. Filling the instance's slot value.

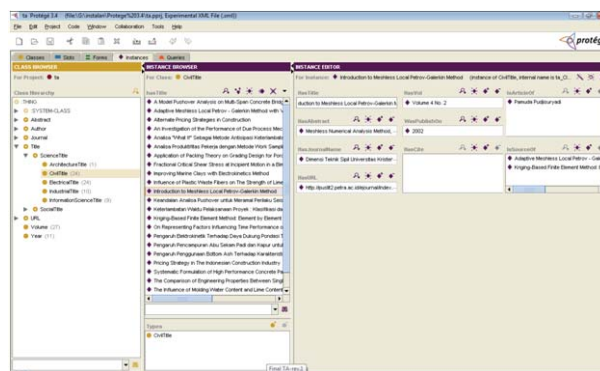Figure 5 below is a Protege page consists of a class, slot and instance browser [1].

Figure 5. Protégé Page

D.   Interface Design

Interface design is used to upload and parse the XML file into database MySQL by a domain expert. The following is a quotation of the ontology model in the form of XML.

```
<simple_instance>
                <name>ta_Class30103</name>
                <type>CivilTitle</type>
                <own_slot_value>
            <slot_reference>hasAbstract</slot_reference>
<value value_type="simple_instance">ta_Class30110</value>
                </own_slot_value>
                <own_slot_value>
            <slot_reference>isArticleOf</slot_reference>
<value value_type="simple_instance">ta_Class30104</value>
                </own_slot_value>
                <own_slot_value>
            <slot_reference>hasJournalName</slot_reference>
<value value_type="simple_instance">ta_Class23</value>
                </own_slot_value>
                <own_slot_value>
            <slot_reference>hasTitle</slot_reference>
<value  value_type="string">Application of Packing Theory on
        Grading Design for Porous Asphalt Mixtures</value>
                </own_slot_value>
                <own_slot_value>
            <slot_reference>hasURL</slot_reference>
<value value_type="simple_instance">ta_Class30109</value>
                </own_slot_value>
                <own_slot_value>
```

```
                    <slot_reference>hasVol</slot_reference>
<value value_type="simple_instance">ta_Class30038</value>
                    </own_slot_value>
                <own_slot_value>
            <slot_reference>isSourceOf</slot_reference>
<value value_type="simple_instance">ta_Class30100</value>
                    </own_slot_value>
                <own_slot_value>
        <slot_reference>wasPublishOn</slot_reference>
<value value_type="simple_instance">ta_Class20036</value>
                    </own_slot_value>
            </simple_instance>
```
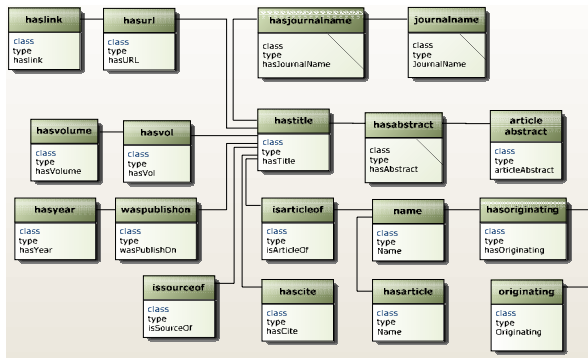
Figure 6 below shows the parsing result in the form of database which contain 18 tables.

Figure 6. Database and its relationship

Figure 7 below shows the design of the login page and Figure 8 shows the design the of the upload and parsing page.
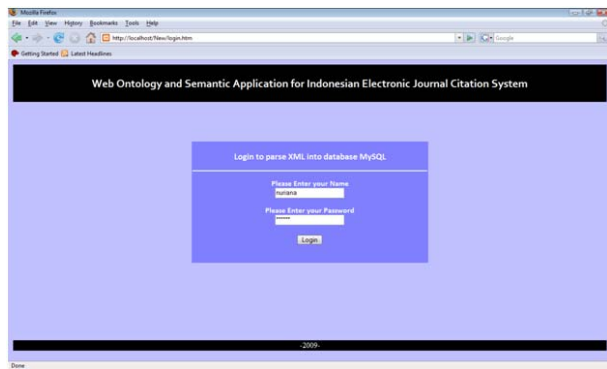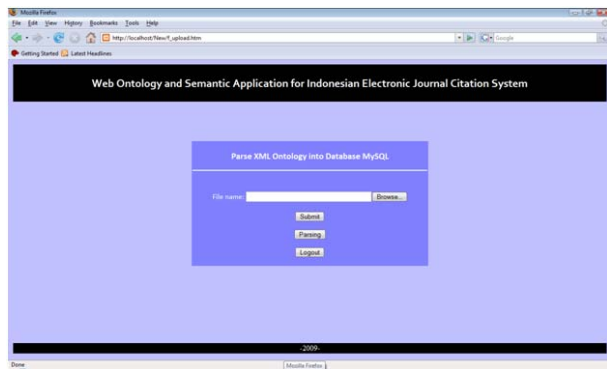

Figure 7. Design of the login page

Figure 8. Design of upload and parsing page.

E. Portal Design

Web portal presents information from diverse resources in a unified way. The portal design provides information about the components available on a page as shown in Figure 9 below.
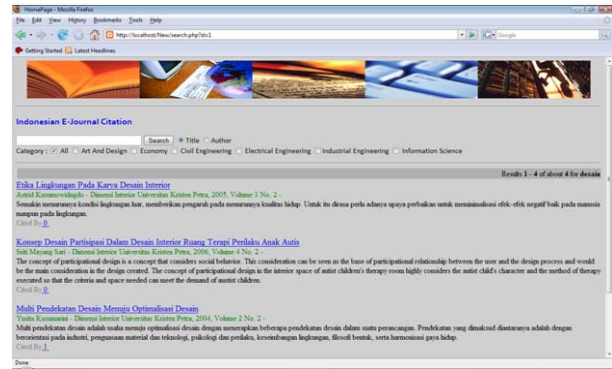

Figure 9. Search Results

On the page, user will find the text box and search button to search. Search is supported by the radio button options menu such as Title and Author with multiple checkbox in the category All, Art and Design, Economy, Civil Engineering, Electrical Engineering and Information Science. When user selects the radio button *Title*, queries will be directed to the table *hasTitle* that result from parsing XML file. When user select the radio button *Author*, queries will be directed to the table *is ArticleOf*. Checkbox will direct queries to the class in the ontology *Title* that are on the table *hasTitle*. If the checkbox Art and Design is selected, system will display only the articles from Art and Design category, which is also valid for the other checkboxes.

The search results will be display the following result:
- The first line will display the article title.
- The second line will display the author of the article, journal name, year and volume of articles published.
- The third line will display abstraction of the article.
- The fourth line will display the number that article in the first line had been cited, If the number is pressed, it will display the article cited in it.

F. Activity Diagram

The activity diagram shows how the system process the request and how the user provides input and the system respond to these inputs. The complete systems are presented in Figure 10 and 11. Figure 10 is a diagram of domain expert or librarian to the web server to upload and parse, while Figure 11 shows a diagram of how user performed web search from the system.
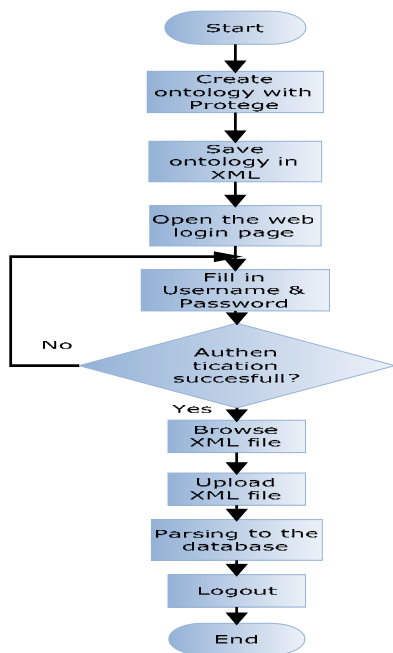
Figure 10. Flow chart of domain expert to upload and parse XML file.

After the modeling is made in Protege, it will be uploaded and parsed to MySQL database. In this case domain expert or librarian must first do the authentication, to limit the access to eligible users to update MySQL database. Users opens a web page and enter the username and password. If the user can open the web page successfully, user can upload and parse the XML file.
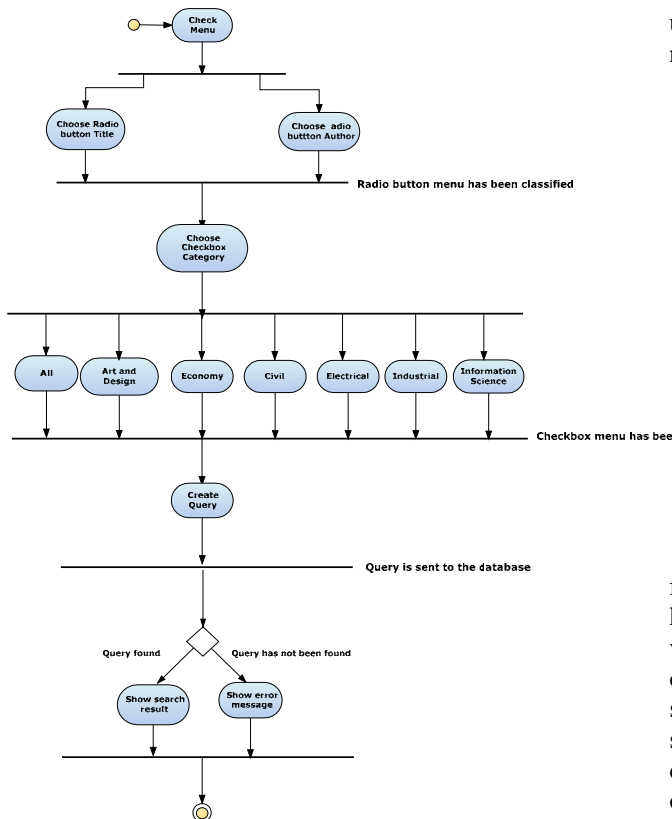

Figure 11. Activity Diagram of the System

On the web page there are several main search menus based on title indicated by radio button Title and based on Author indicated by radio button Author.

On selecting the radio button, users are exposed to the category selection which is shown with a checkbox. Category consists of all, art and design, economy, civil engineering, electrical engineering, industrial engineering and information science.

Users will put the keyword in the textbox and press the search button in performing the search. When the search result is found then the information will appear on the screen.

IV. TESTING AND ANALYSIS

A. User Testing

Testing is conducted by ten users by filling questionnaires. The questionnaire is a set of questions related to product performance system. There are two types of users, namely the user which acts as a librarian and the user performing a search. The measurement using the scale $1 - 4$ of that statement given that represent 1 = extremely disagree, 2 = disagree, 3 = agree, 4 = extremely agree. The user responses have been statistically evaluated for the fact that we are using 95% confidence interval, and show the maximum and minimum value for testing conducted by ten users.

B. Domain Expert or Librarian as the user.

Test has been conducted to determine the effectiveness of using Protégé as an ontology tool and whether it can be used for our electronic journal citation system easily. The result of test is shown in Figure 12.
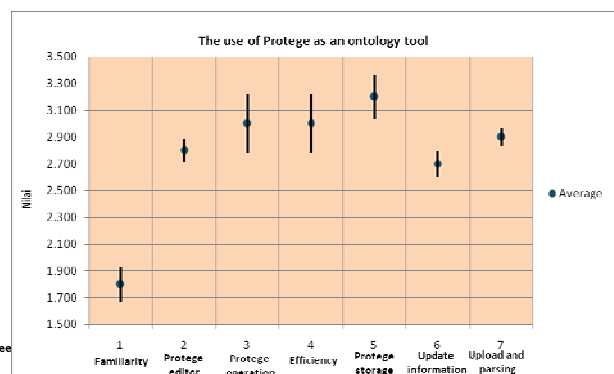

Figure 12. Graphic Feedback Protégé as Ontology Tool

The response to the question about the familiarity in Protégé shows that only a very few users know about Protégé as an ontology tool. The average value obtained in the first question only on scale of 1.8 out of 4. This is due to the fact that not many ontology systems have been developed in Indonesia. The trend shows that information is not semantically stored. In order to ensure that the information is stored semantically or information is correct to be regarded as a more concrete knowledge, we can used ontology as the base of modeling. Protégé has been selected as a tool to perform

the function of ontology. By using Protégé, knowledge models are developed in the form of ontology class hierarchy, slots and instance. The Ontology model is able to describe the information related to the electronic journal,

The second question refers to the introduction of tabs as the embodiment of the components, namely class, slot and instance. The average value obtained from a question-tab in a tab ontology editor is easy to understand, the scale value is 2.8. Eighty (80) % of the respondents consider that the tab is easy to understand but 20% of respondent argued that the tabs are difficult to understand. This is due to the fact that some users do not understand the connection between the components.

The third question is about the operation the Protégé against the user's role as a librarian. Before testing, the user has been given a manual and guidance. User continues with entering a new article in the electronic journal by using model ontology that had been created. The average value obtained is in the scale of 3, which means that users agree that the Protégé is easy to run. The system is easy to be used for the friendliness of the user interface. The system works well and help manual has been provided so the users do not need to follow special training.

The fourth question is related to the third and sixth question, the test of efficiency in the entering data. Data enter is conducted manually, but is not complicated for the user, because the form is created as clearly as possible so users do not need to repeatedly input the data and the data is classified in each class and subclass. The system prevents duplicate data although the data is used many times. The average value obtained is on the scale of 3 which indicates that the user agree with the entering data method through a Protégé.

The fifth question is about the storage model. The average obtained was on the scale of 3.2 that Protégé help in the process of storage for citation system. Modeling in this application is stored in XML documents. Besides XML, the formats provided by Protégé are OWL, RDF and HTML [9, 10, 11]. Protégé provides ease of plug and play which makes it flexible for ontology development.

The sixth question shows that 70% of respondent agree that the update of new article is relatively simple and 30 % of respondents considers that it is still difficult to enter a new article. Users are required to examine and consider the references in detail whether an article cite other article in electronic journal. The average value for this question has a scale of 2.7.

The last test is conducted by a domain expert or librarian to upload and parse XML document. This application is an interface between XML document and database MySQl. The average value obtained is 2.9, which means that the system is easy to be used and helpful in terms of its web interface application.

C. Web User as The Tester
In the second part, the question has been provided to know the use of web as a search engine to search citation

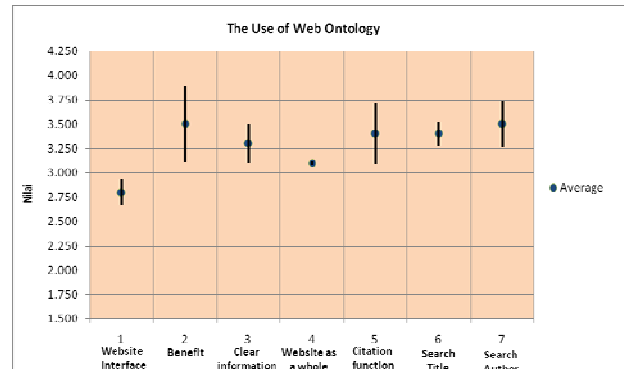in electronic journal. The result of test is shown in Figure 13.

Figure 13. Graphic Feedback Web Interface

Questionnaires are used to evaluate customer satisfaction of the application. The first question is about the attractiveness of website. The second question is about the usefulness of the system to get the information. The third question is the about how clear is the information to the user. The fourth question, website could work smoothly. The fifth question, system could show its function to search for article's citation. The sixth question, system could show its function to search article based on title. The seventh question, system could show its function to search article based on author. The average result is more than 3 which means that users are relatively satisfied with the application. The system performs its function as a medium for searching based on the article title, author and category which have been modeled in the ontology. The search results are very useful and clear information has been displayed which fulfill the requirement previously made in the development of this ontology based system which is to answer questions from a domain.

V. CONCLUSIONS

Based on implementation and testing of the system, some conclusions obtained are as follows:

1. Web ontology development and semantic application can be implemented in accordance with the design.
2. Ontology modeling can be used as the database for Indonesian electronic journal citation system.
3. Ontology modeling is an approach of semantic modeling. Ontology modeling gives the meaning from symbols, name class, slot and instance so that the application can describe the semantic of all document content.
4. The benefits of semantic approach apparently are the shorter time to get the information.
5. The election of Protégé as a tool because of the open source product, storage in the form of XML document and the availability of the manual book and plug-in visualization called TGViz.
6. Protégé is relatively simple ontology tool which can be implemented in the electronic journal citation system even though the user is not familiar with the

tool. The average scale is 2.94 and has a percentage of 73.5 %.

7. The web as the user interface work properly is assessed by the users. The average value obtained in the scale is over 3.

8. From the citation result, 90 articles from all journal categories of Udayana University and Petra Christian University have relationship between the articles. The result shows that citation comes from the same author as the development of previous scientific paper.

ACKNOWLEDGMENT

REFERENCES

[1] Eliza Sachs, "Getting Started with Protégé-Frames", 2006, http://protege.stanford.edu. Accessed on March 10th 2009.

[2] Matthew Horridge, Holger Knublauch, Alan Rector, Robert Stevens, Chris Wroe, "A Practical Guide to Buiding OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools Edition 1.0". http://protege.stanford.edu. Accessed on March 8th 2009.

[3] Natalya F. Noy, Deborah L. McGuiness, "Ontology Development 101: A Guide to Creating Your First Ontology", p. 1, http://protege.stanford.edu. Accessed on May 17th 2009.

[4] Soren Auer, "Powl – A Web Based Platform for Collaborative Semantic Web Development", pp. 2, www.powl.com Accessed on May 20th 2009.

[5] "DL Query Tab", p. 1, http://protegewiki.stanford.edu/index.php/DLQueryTab . Accessed on May 20th 2009.

[6] Leigh Dodds, "Introducing SPARQL: Querying the Semantic Web", p. 1, http://www.xml.com. Accessed on May 17th 2009.

[7] Dave Beckett, Jeen Broekstra, "SPARQL Query Results XML Format", p. 1, http://www.w3.org. Accessed on May 15th 2009.

[8] Radoslaw Oldakowski, Christian Bizer, Daniel Westphal, "RAP: RDF API for PHP" , p. 1, http://www.w3.org. Accessed on May 15th 2009.

[9] Holger Knublauch, Olivier Dameron, Mark A. Musen, "Weaving the Biomedical Semantic Web with the Protégé OWL Plugin", p. 1, http://protege.stanford.edu. Accessed on May 15th 2009.

[10] Robert Baumgartner, Nicola Henze, Markus Herzog, "The Personal Publication Reader: Illustrating Web Data Extraction, Personalization and Reasoning for The Semantic Web", 2005.

[11] "OWL, Web Ontology Language, W3C Recommendation", p. 1, http://www.w3.org. Accessed on May 15th 2009.