ComTech: Computer, Mathematics and Engineering Applications, 10(1), June 2019, 1-7
DOI: 10.21512/comtech.v10i1.4962

P-ISSN: 2087-1244
E-ISSN: 2476-907X

Expert System Development for Course Enrollment Process Using Ripple Down Rules in a University in Surabaya

Agus Cahyo Nugroho

Information Technology, Creative Industry Faculty, Ciputra University
UC Town, Citraland CBD Boulevard, Surabaya 60219, Indonesia
agus.nugroho@ciputra.ac.id

Received: 30th August 2018/ **Revised:** 12th October 2018/ **Accepted:** 4th December 2018

How to Cite: Nugroho, A. C. (2019). Expert System Development for Course Enrollment Process Using Ripple Down Rules in a University in Surabaya. *ComTech: Computer, Mathematics and Engineering Applications, 10*(1), 1-7. https://doi.org/10.21512/comtech.v10i1.4962

***Abstract* -** The research aimed to develop expert system for helping lecturers to decide which courses to be taken by students in the next semester. The expert system contained specific pieces of knowledge to solve specific problems involved in the forms of system development and maintenance. This expert system was in the form of a website using the PHP programming language and MySQL database. The researcher used the Ripple Down Rules (RDR) method to identify the courses by putting forward the questions so that the system could decide which courses the students should take in next semester. The result shows that this web-based expert system can identify which courses that the students have to enroll after the students have answered questions generated by the system. The available data on the courses in the system adapt to the rules, so it is in line with the enrolled courses.

Keywords: expert system, course enrollment, Ripple Down Rules (RDR)

I. INTRODUCTION

Information can be found in various forms or generated from some available knowledge. Text, images, audio, and video are different forms of media from which information can be accessed. The role of information technology is to invent and devise tools to store and retrieve this information. Along with the development of technology, a system is developed to adopt the way people think and the processes involved therein.

Expert systems can be defined as tools for generating information from knowledge. It is capable of acting in accordance with a human reasoning process, giving similar advice, and making similar decisions to what human expertise may do. In short, it can gather vast numbers of information and experience from multiple experts of numerous disciplines and provide valuable recommendations to users. Expert systems can become beneficial assistants to human decision makers.

Nowadays, the expert system with another method faces many problems in the forms of system development and maintenance. These problems occur because the knowledge acquisition from experts is only performed at the early stages of system development. It makes the system incapable of solving new rising problems in the later stages. It is because there will be no rules or facts that will serve as the base for the solutions of these problems. Another issue with the expert system with another method is that it cannot change the existing rules or facts. It can damage the base rules that are already in the system.

Responding to these issues, the researcher attempts to study how the Ripple Down Rules (RDR) method can be employed to solve the mentioned issues. RDR is a new knowledge acquisition method which a system can develop new rules on its own. Then, the new rules remain to be in accordance with the existing base rules. Moreover, if an expert assumes that the existing knowledge no longer matches what they intend, RDR will allow the expert to alter or update the existing knowledge. Consequently, it will allow the system to obtain the latest expert knowledge.

The application of RDR method has been conducted in clinical decision support system as the system needs to predict and complete missing information for generating appropriate recommendations by Hussain, Hassan, Sadiq, Kang, and Lee (2018). They extended RDR method that identified the missing information regarding key facts by analyzing similar previous patient cases.

Then, Anam, Kim, Kang, and Liu (2015) stated that machine learning approaches could learn their model by using the data. However, they were static, so they could not be modified to reflect the domain data changes. Inversely, the knowledge engineering approaches needed domain experts. It could be modified by reflecting the domain data changes. To exploit the advantages of both approaches and reduce the limitations, they proposed a hybrid approach called as Hybrid – RDR. They combined a machine learning algorithm with RDR, an incremental knowledge engineering approach.

RDR and the Multiple Classification Ripple Down Rules (MCRDR) variants have enjoyed excellent research reviews and commercial outputs in the last two or more decades (Richards, 2009). For example, RDR and MCRDR have been used across a diversity of research and application areas: telehealth (Han *et al*., 2013); legal text citation (Galgani, Compton, & Hoffmann, 2015), and flight control systems (Shirazi & Sammut, 2008). As for the speed of development and implementation, it has been shown that the MCRDR-backed knowledge base method is closely aligned with the agile software development approach (Han, Yoon, Kang, & Park, 2014).

The reason why the researcher chooses the course enrollment process topic is that the researcher has revealed the factors that cause academic procrastination. Patrzek, Sattler, van Veen, Grunschel, and Fries (2015) showed that students engaged in academic procrastination because of their personality, competence, affection, cognitive, physical and mental health, perception against the characteristics of a task, and personal factors. Personal factors included the lack of social support, continual stress, and critical events that they had experienced throughout their lives. The factors associated with schools such as teacher quality and school conditions were also included.

The research on academic procrastination by students has been overwhelmingly revealing. Zeenath and Orcullo (2012) found that among 287 students in Malaysia who were facing a test, 80% of the students had engaged in procrastination. It showed that 32,5% of the students had been preparing for the exam for two weeks before the exam, 20% of the students had been preparing for the exam in the last minutes after finishing other assignments and 27,5% did not prepare for the exam at all. Exam preparation had also been found to depend on the students' moods with the consequence that a majority of the students had engaged in procrastination during the preparation for an exam.

Patrzek *et al*. (2015) suggested that both internal and external factors caused academic procrastination. The internal factors include many types. First, it was personality-related factors such as negative self-image, avoidance, and perfectionism. Second, there were factors related to the competence of the students such as low self-regulation, lack of time management skills, low learning skills, and lack of knowledge. Third, there were affective factors such as anxiety, frustration, and depression. Fourth, it was cognitive factors such as worries, the fear of failure, and irrational beliefs. Fifth, there were factors related to learning history such as learning behavior and negative learning experience. Sixth, it was related to physical and mental health such as illness and impairment. Seventh, it was related to the perception of the characteristics of the assignment such as the difficulty level, the workload, and assignments that were not interesting and unpleasant. On the other hand, external factors included personal factors such as the lack of social support, continual stress, critical events taking place in one's life, and factors related to the school such as teacher quality and school conditions.

Patrzek *et al.* (2015) stated that academic procrastination had both positive consequences and negative consequences. The positive consequences of academic procrastination included a reduction in stress level, additional time for socialization, needs adjustment, and increased opportunity to enjoy life in school. On the other hand, the negative consequences of procrastination academic included low self-esteem, affective consequences such as anxiety, dissatisfaction, depression, declining motivation, and stress, and school-related consequences which were accumulated and became increasingly burdensome. Those resulted in low learning achievement and possibly dropping out of school.

Course enrollment process included researching available courses. It is usually done by students who intend to share their academic course plans with their academic supervisors. An academic supervisor with a small group of supervised students and a few courses to teach will find it easier to manage the students consultation schedule. However, an academic supervisor with a large number of supervised students and many courses to teach will find it harder to manage the students' consultation schedules.

From the background explained, the researcher has attempted to develop an expert system for course enrollment process using RDR method.

II. METHODS

The first step in this research is data gathering and requirement analysis. The source of data gathering is the curriculum. The second step is designing a database and tables. The third step is database implementation in MySQL. The fourth step is web programming and design using PHP and Bootstrap.

RDR that the researcher uses has been the knowledge acquisition technique initially. It allows the compilation of clinical remarks or the interpretation of lab reports to help doctors provide referrals. In its development, RDR is a strategy in developing a system. When the system does not respond correctly, the changes are needed without affecting the system competence. The changes should be easily and quickly. However, the difficulty in making these changes should not increase. The system should be maintained even though the system evolves.

The method used in this research is forward chaining. Forward chaining is an inference method beginning with compiling known facts and new facts using rules that have a corresponding premise with other known facts. This process continues until a conclusion is achieved or until no rules correspond to those known facts (Durkin, 1994).

III. RESULTS AND DISCUSSIONS

This research only based on application development without user testing. User testing will be held in the next development research when application implemented in real world.

Inference engine developed in this expert system uses a set of rule to generate diagnosis results based on data input by the users. A forward chaining inference process by Durkin (1994) is used as the reasoning approach. This approach automatically matches facts with the patterns to determine which rules match them. Forward chaining in Figure 1 starts with the available data and uses inference rules to extract more data until the objective is met. Once a rule is found, the engine will infer a conclusion. This iterative process will continue until the objective is met.

The rules contained in this method are tested one by one in particular sequences. As each rule is tested, the expert system will evaluate whether a condition is true or false. If the condition is true, the rule will be stored and tested. Otherwise, if the condition is false, the rule will not be saved. This process will be performed over and over until all of the rules have been tested in various conditions.

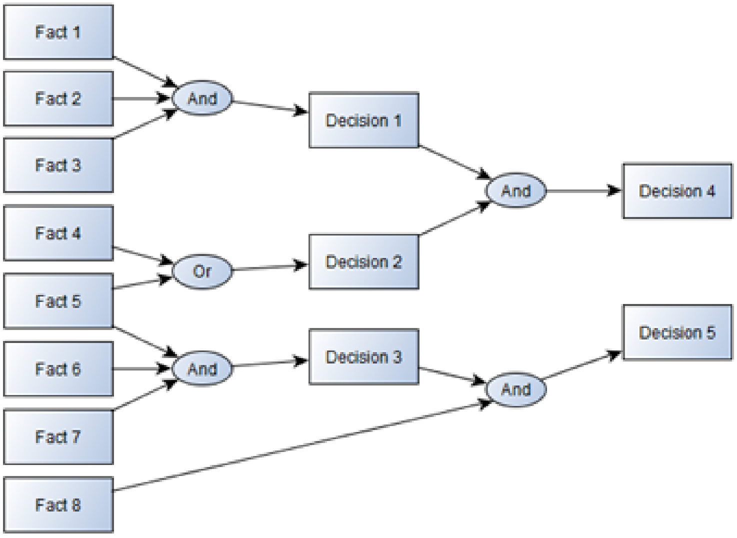


Figure 1 Forward Chaining
(Source: Bratko, 2001)

If in its identification, a case is found to be true, the expert will only need to enter true rather than false facts. Then, the system will form a rule solely based on the facts given by the expert. On top of that the rule, there will be marked as a rule to be updated. Updating a rule will result in the rule being treated as an addition to the existing exception rules without any alternation to the order and the quality of the already placed rules. If a rule has been updated, the user will see the updated view. The example of an implementation of the RDR method can be seen as follows.

The initial rules formed in the system:

1. IF Algorithm and Programming
2. AND Indonesian
3. AND Introduction to IT
4. AND Operating System
5. AND Essentials of Entrepreneurship
6. THEN Student is a Second Semester student

The expert can change an old subject in the system with a new subject. For example, the expert changes the previous Indonesian course with an English course. Several ways are used to implement the change of the course performed by the expert.

First, the system is going to show everything related to the course that is changed.

Subject Code: P001
Subject Name: Semester 2
Subject Code: G001
Subject Name: Algorithm and Programming
Subject Code: G002
Subject Name: Indonesian
Subject Code: G003
Subject Name: Introduction to IT
Subject Code: G004
Subject Name: Operating System
Subject Code: G005
Subject Name: Essentials of Entrepreneurship

Second, the system will ask which subject that will be changed in this system.

Subject Code: P001
Subject Name: Semester 2
Subject Code: G002
Old Subject Name: Indonesian

Third, the system will also ask for the addition of a new rule to the expert system.

New Subject Name: English

Fourth, the system will track the associated rule base with Semester 2 and Indonesian and change the old subject into English. After that, the system will order the newly formed rule:

1. IF Algorithm and Programming
2. AND English
3. AND Introduction to IT
4. AND Operating System
5. AND Essentials Entrepreneurship
6. THEN The Semester is Semester 2

The flowchart in Figure 2 describes the flow of the created working. When a user or an expert chooses the consultation menu, the system will ask different questions that will be displayed in accordance with the existing rule base. The answer from the user will be entered to the working memory. Then, the system will check the answers with the rules contained in the rule base.

If the rules and the answers in the working memory are matched. The next question will be displayed in accordance with the existing rule base. When the user has answered the question, the system will display the results based on the answers entered by users. However, if the rules and the answers are not matched, the default output will be displayed to the user.

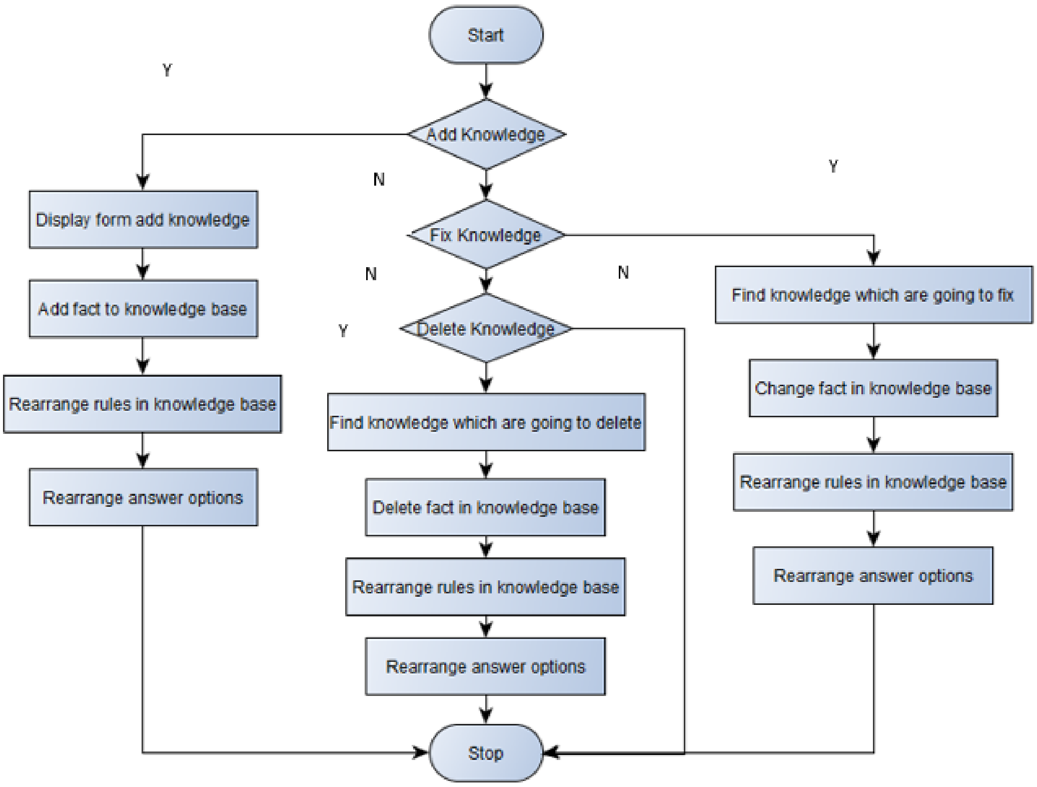

Start
Add Knowledge
Y
N
Display form add knowledge
Add fact to knowledge base
Rearrange rules in knowledge base
Rearrange answer options
Fix Knowledge
Y
N
Delete Knowledge
Y
N
Find knowledge which are going to delete
Delete fact in knowledge base
Rearrange rules in knowledge base
Rearrange answer options
Find knowledge which are going to fix
Change fact in knowledge base
Rearrange rules in knowledge base
Rearrange answer options
Stop

Figure 2 A Consultation Process in the System

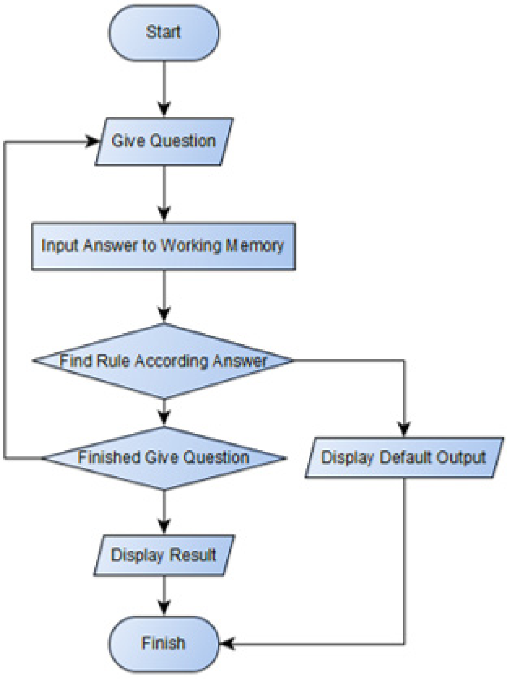


Figure 3 RDR Implementation Flowchart

Figure 3 describes how the RDR method will work in this expert system. When an expert has successfully logged in the system, the expert will be directed into a special consultation menu for an expert user. This menu has several buttons that can be used by the expert to go to the next menu. The RDR method will be activated when the expert chooses the add, fix, or delete knowledge buttons. For example, if the expert chooses the add menu, the system will display the form for adding knowledge and add the inserted knowledge into the system.

Next, if the expert chooses the fix or delete knowledge menu, the system will search for old knowledge that will be changed and change that knowledge. New knowledge can be created by entering it. Then, the system will build a new rule based on the knowledge entered by the expert. The system will convert it into a new rule form and insert it into the base rules. Then, the system will create a set of repeated rules that are in the rule base.

Knowledge base contained in this expert system keeps the facts related to taken subjects and its sum of Satuan Kredit Semester (SKS – University Credit Unit), and credit points that will later be used for a conclusion. These facts will be obtained from the results of interviews with experts and other sources such as books, journal articles, and general articles. After that, they will be translated into computer language and recorded into the database.

The facts in this expert system can be classified into five categories. Those are semester, taken subject, sum of SKS, sum of credit point, and next semester. The data can see in Tables 1−5.

Table 1 presents the relationship between code of semester with which semester it belongs to. This is a reference table. Table 2 is the relationship between the code of subjects in which semester and what subjects that have been taken. Table 3 shows the relationship between codes and the ideal sum of SKS for each semester. Table 4 presents the relationship about code and the ideal sum of credit point in each semester. Table 5 is the relationship of codes and the next semester position. Furthermore, these facts will be used to build decision trees and base rules to reach the solution as required by the system.

Table 1 Code_sems and Semester

No	Code_sems	Semester
1.	B001	Semester 1
2.	B002	Semester 2
3.	B003	Semester 3
4.	B004	Semester 4
5.	B005	Semester 5
6.	B006	Semester 6
7.	B007	Semester 7
8.	B008	Semester 8

Table 2 Code and Subject Taken

No	Code_sub	Code_sems	Taken_subject
1.	G001	B001	Indonesian
2.	G002	B001	Algorithm and Programming
3.	G003	B001	Introduction to IT
4.	G004	B001	Operating System
5.	G005	B001	Essentials Entrepreneurship
6.	G006	B002	Civics
7.	G007	B002	Programming Techniques
8.	G008	B002	Web Programming
9.	G009	B002	Basic Mathematics
10.	G010	B002	Multimedia
11.	G011	B003	Being an Indonesian
12.	G012	B003	Mobile Apps Development
13.	G013	B003	Web Development
14.	G014	B003	Database
15.	G015	B003	Discrete Mathematics
16.	G016	B003	Game Design
17.	G017	B004	Pancasila
18.	G018	B004	Advanced Mobile Apps Development
19.	G019	B004	Technopreneurship
20.	G020	B004	Computer Graphics
21.	G021	B004	Machine Learning
22.	G022	B004	University Elective 1
23.	G023	B004	Internet Of Things
24.	G024	B005	Religion
25.	G025	B005	Artificial Intelligence
26.	G026	B005	Product Development
27.	G027	B005	Statistics and Probability
28.	G028	B005	Elective 1
29.	G029	B005	University Elective 2
30.	G030	B006	Computer Network
31.	G031	B006	Advanced Product Development
32.	G032	B006	Research Methodology
33.	G033	B006	Elective 2
34.	G034	B006	University Elective 3
35.	G035	B007	Internship
36.	G036	B008	Applied Ethics
37.	G037	B008	Final Project
38.	G038	B008	Elective 3

Table 3 Code and Sum of SKS

Code_sum_sks	Sum_sks
NH001	18
NH002	36
NH003	57
NH004	75
NH005	95
NH006	116
NH007	130
NH008	144
NH009	>144

Table 4 Code and Credit Point

Code_credit_point	Sum_credit_point
H001	15
H002	30
H003	45
H004	60
H005	75
H006	90
H007	105
H008	120

Table 5 Code and Next Semester

Code_dn	Next_semester
P001	Semester 2
P002	Semester 3
P003	Semester 4
P004	Semester 5
P005	Semeser 6
P006	Semester 7
P007	Semester 8

The system use case diagram can be seen in Figure 4. Admin has four main features after successfully logging in the page. Those are Manage semester, Manage subject, Manage SKS, and Manage credit point. The other users do not have to log in to use consultation menu.

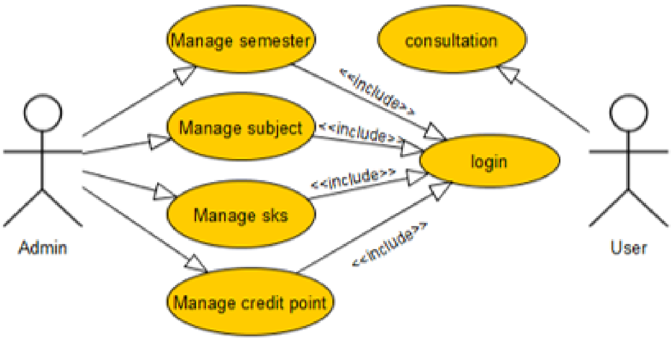
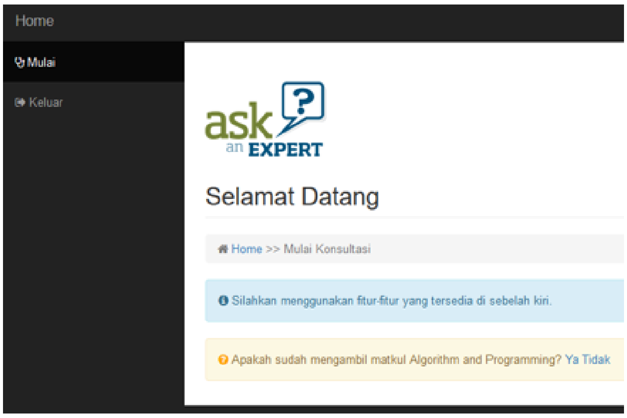


Figure 4 System Use Case Diagram

The consultation menu will be displayed when the expert or the user has pressed the start button on the expert menu panel or the user menu panel as seen in Figure 5. This menu is used to consult the system on problems encountered in the course enrollment process. The questions are produced by system in Yes or No question. Thus, the user only needs to choose one of the answers. If it is Yes, there will be next question produced by the system. If it is No, there will also be question produced by the system. Based on user's answer in the system, it will give the final conclusion. It aims to identify which course that students should take the next semester. The system is designed for package subjects in a semester.



Selamat Datang

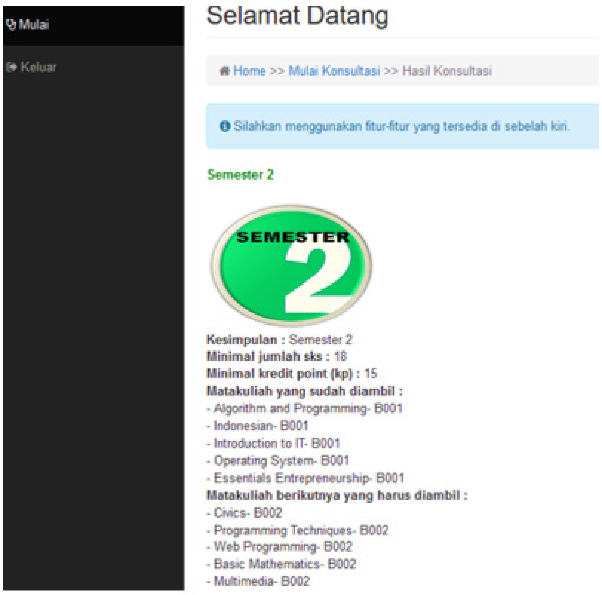
Home >> Mulai Konsultasi

Silahkan menggunakan fitur-fitur yang tersedia di sebelah kiri.

Apakah sudah mengambil matkul Algorithm and Programming? Ya Tidak

Figure 5 Consultation Menu

Figure 6 shows the menu that will appear after the consultation process has been done in a system. This menu contains an explanation on the students' position in the current semester, taken courses, the total number of academic credits earned, the total credit points, and the courses in the next semester that they may have to be enrolled in. Then, Figure 7 shows the summary result of the consultation.



Selamat Datang

Home >> Mulai Konsultasi >> Hasil Konsultasi

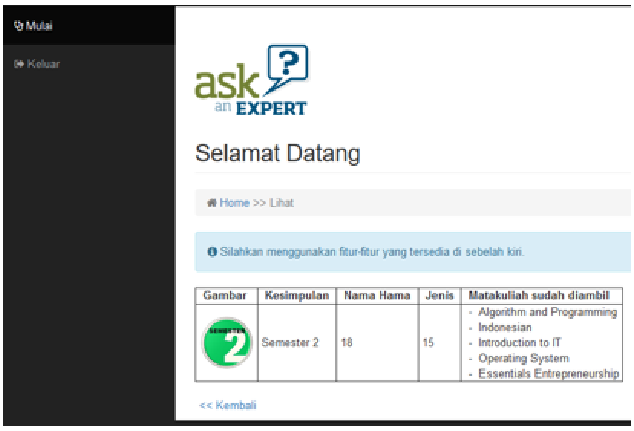
Silahkan menggunakan fitur-fitur yang tersedia di sebelah kiri.

Semester 2

Kesimpulan : Semester 2
Minimal jumlah sks : 18
Minimal kredit point (kp) : 15
Matakuliah yang sudah diambil :
- Algorithm and Programming- B001
- Indonesian- B001
- Introduction to IT- B001
- Operating System- B001
- Essentials Entrepreneurship- B001

Matakuliah berikutnya yang harus diambil :
- Civics- B002
- Programming Techniques- B002
- Web Programming- B002
- Basic Mathematics- B002
- Multimedia- B002

Figure 6 Consultation Result Menu



Selamat Datang

Home >> Lihat

Silahkan menggunakan fitur-fitur yang tersedia di sebelah kiri.

Gambar	Kesimpulan	Nama Hama	Jenis	Matakuliah sudah diambil
	Semester 2	18	15	- Algorithm and Programming - Indonesian - Introduction to IT - Operating System - Essentials Entrepreneurship

<< Kembali

Figure 7 Consultation Result

IV. CONCLUSIONS

The conclusion can be drawn based on the implementation result and the expert system analysis using the RDR for course enrolment process. The expert system is a solution that will help lecturers to direct the students regarding what subjects they have to take next semester. RDR method can be used to process settings and maintain the expert system. This method is capable of rearranging the rules contained in rules based on facts added, changed or fixed by an expert in this system.

Moreover, the expert system can maintain the consistency of the facts and rules in the processes of addition, fixing, and deleting. This system is expected to help knowledge acquisition from expert to system. Thus, the system can have the newest knowledge from the expert. Beside that, updating rule also guarantees the new knowledges from expert as it does not have to break the basic rules of the system. It is developed by using bootstrap CSS. It has a flexible layout in a web browser and mobile browser. Thus, it is easier for users to use this system in the web browser or the mobile application.

The research limitation is that the system is still designed for package subjects in a semester. Moreover, there is no history for updating rule. Therefore, it is suggested for the future research to develop the custom subjects in each semester and history to update the rule.

REFERENCES

Anam, S., Kim, Y. S., Kang, B. H., & Liu, Q. (2015). Schema mapping using hybrid ripple-down rules. In *38th Australasian Computer Science Conference (ACSC 2015)* (Vol. 159, pp. 17-26).

Bratko, I. (2001). *Prolog programming for artificial intelligence*. Pearson Education.

Durkin, J. (1994). *Expert systems: Design and development*. New York: Macmillan.

Galgani, F., Compton, P., & Hoffmann, A. (2015). Lexa: Building knowledge bases for automatic legal citation classification. *Expert Systems with Applications, 42*(17-18), 6391-6407.

Han, S. C., Mirowski, L., Jeon, S. H., Lee, G. S., Kang, B. H., & Turner, P. (2013). Expert systems and home-based telehealth: Exploring a role for MCRDR in enhancing diagnostics. In *International Conference, UCMA, SIA, CCSC, ACIT 2013* (Vol. 22, pp. 121-127).

Han, S. C., Yoon, H. G., Kang, B. H., & Park, S. B. (2014). Using MCRDR based agile approach for expert system development. *Computing, 96*(9), 897-908.

Hussain, M., Hassan, A. U., Sadiq, M., Kang, B. H., & Lee, S. (2018). Missing information prediction in ripple down rule based clinical decision support system. In *International Conference on Smart Homes and Health Telematics* (pp. 179-188).

Patrzek, J., Sattler, S., van Veen, F., Grunschel, C., & Fries, S. (2015). Investigating the effect of academic procrastination on the frequency and variety of academic misconduct: A panel study. *Studies in Higher Education, 40*(6), 1014-1029.

Richards, D. (2009). Two decades of ripple down rules research. *The Knowledge Engineering Review, 24*(2), 159-184.

Shirazi, H., & Sammut, C. A. (2008). Acquiring control knowledge from examples using ripple-down rules and machine learning. *Iranian Journal of Science and Technology, 32*(B3), 295-304.

Zeenath, S., & Orcullo, D. J. C. (2012). Exploring academic procrastination among undergraduates. *International Proceedings of Economics Development & Research, 47*(9), 42-46.