DEVELOPMENT OF CASE-BASED RECOMMENDATION SYSTEM FOR COURSE SELECTION IN A UNIVERSITY CONTEXT

By ZEHRA BEGÜM EROĞLU

SEPTEMBER 2018

DEVELOPMENT OF CASE-BASED RECOMMENDATION SYSTEM FOR COURSE SELECTION IN A UNIVERSITY CONTEXT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF
ÇANKAYA UNIVERSITY

BY
ZEHRA BEGÜM EROĞLU

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
COMPUTER ENGINEERING
DEPARTMENT

SEPTEMBER 2018

Title of the Thesis: **Development of Case-based Recommendation System for Course Selection in a University Context**

Submitted By **Zehra Begüm EROĞLU**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University.

Prof. Dr. Can ÇOĞUN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Erdoğan DOĞDU
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Murat SARAN
Supervisor

Examination Date: 14.09.2018

Examining Committee Members

Assist. Prof. Dr. Gökhan ŞENGÜL Atılım Univ.

Prof. Dr. Erdoğan DOĞDU Çankaya Univ.

Assist. Prof. Dr. Murat SARAN Çankaya Univ.

STATEMENT OF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Zehra Begüm EROĞLU
Signature:
Date: 14.09.2018

ABSTRACT

DEVELOPMENT OF CASE-BASED RECOMMENDATION SYSTEM FOR COURSE SELECTION IN A UNIVERSITY CONTEXT

EROĞLU, Zehra Begüm
M.Sc., Department of Computer Engineering
Supervisor: Assist. Prof. Dr. Murat SARAN

SEPTEMBER 2018, 45 pages

Nowadays, recommendation systems that guide people for their choices are one of the most widely used software technologies. Especially in the field of education, this technology can be used to make course selection more efficient for students. In this thesis, a recommendation system for the course selection of the students was developed and the accuracy of the recommendations were tested. In the system developed within the scope of this study, Case-Based Reasoning (CBR) method was applied by using the real data of the previous students. While preparing the proposals, the data of students who take similar courses in their curriculum are used. The CBR method presented the suggestions of the courses to be taken by the students and the success rate of the system was calculated by comparing the appropriateness of the relevant suggestions with the lessons actually taken. According to the results of this study, the success rate of the suggestion system is 69%.

Keywords: Case-based reasoning, recommendation system.

ÖZ

VAKA TEMELLİ AKIL YÜRÜTME YÖNTEMİ İLE ÜNİVERSİTE BAĞLAMINDA DERS SEÇİMİ İÇİN ÖĞRENCİ ÖNERİ SİSTEMİNİN GELİŞTİRİLMESİ

EROĞLU, Zehra Begüm
Yüksek Lisans, Bilgisayar Mühendisliği Anabilim Dalı
Tez Yöneticisi: Dr. Öğr. Üyesi Murat SARAN

EYLÜL 2018, 45 sayfa

İnsanlara seçimleri için rehberlik eden öneri sistemleri, günümüzde en çok kullanılan yazılım teknolojilerinden biridir. Özellikle eğitim alanında, bu teknoloji ders seçimini daha verimli hale getirmek için kullanılabilir. Bu tezde, öğrencilerin ders seçimi için bir öneri sistemi geliştirilmiş ve önerilerin doğruluğu test edilmiştir. Bu çalışma kapsamında geliştirilen sistemde, önceki öğrencilerin gerçek verileri kullanılarak Vaka-Tabanlı Akıl Yürütme (VAY) yöntemi uygulanmıştır. Öneriler hazırlanırken, müfredatlarında benzer dersleri alan öğrencilerin verileri kullanılmaktadır. VAY yöntemiyle, öğrenciler tarafından alınacak derslerin önerileri sunulmuş ve ilgili önerilerin uygunluğu gerçekte alınan derslerle karşılaştırılarak sistemin başarı oranı hesaplanmıştır. Bu çalışmanın sonuçlarına göre öneri sisteminin başarı oranı %69'dur.

Anahtar Kelimeler : Vaka-tabanlı akıl yürütme, öneri sistemi.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to Assist. Prof. Dr. Murat SARAN for his supervision, suggestions and encouragement through the development of this thesis. Also, I would like to thank my family to their continued support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CBR	Case-Based Reasoning
DBS	Database Management System
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
MVC	Model – View – Controller
MySql	Php MyAdmin
Php	1.Preformatted Hypertext Page 2.Personal Home Page
URL	Uniform Resource Locator
Wamp	Windows-Apache-MySQL-PHP

CHAPTER 1

INTRODUCTION

1.1. Overview

CBR means presenting a new recommendation to the user's problem based on past experiences. With this recommendation, it has the role of the user while making decisions. The CBR method provides this process in four different ways. These operations are listed in order; retrieve, reuse, revise and finally retrain parts.

The basic idea of this research is that the students need a practical system when they are selecting their courses. Course coordinators give advice and information about the course subjects. This method is most commonly used by students to decide for such a situation. They can make more knowledgeable decisions when they take a new course. Our system which is a recommendation system for students was designed and developed for universities and enables students to choose a course based on the suggestions of previous students. Since the Case-Based Reasoning method (CBR) is closely related to how previous experiences are essential in current students' decisions, we aim to use the CBR in this research and define and verify its parameters.

The CBR approach solves these problems by comparing the past data saved in the extensive database. For example, the CBR approach provides appropriate guidance for students who wish to make course selection. The CBR approach can offer a course suggestion for the upcoming semester for the students, and thus students can progress their studies by following the guidance. Students complete the course selections in accordance with the suggestions offered by the CBR approach. It will be beneficiary if the suggestions are close to the real data. In other words, this system is like a road map for students. The system is aimed to provide the best solution, yet

some best solutions are not the most appropriate solutions for some students. For such a condition, a review must be performed [1].

If a database has a large number of details, it would be difficult to find the right data for students and consequently to offer them suggestions. However, with the CBR approach, it can make it more comfortable and more convenient. Students who regularly register in the course selection system are possible to get the correct suggestion. However, the CBR [2] approach might not be useful for those students who do not regularly attend courses. This kind of web application and similar ones are convenient for this task. It is because with object-oriented web application and the so-called user visual interface the project can be designed smoothly. PHP is a language that can keep this project going on. It aims to create a useful platform visually for the user. The recommendation system developed in this thesis is useful for students to make the right choice during the course selections process. The study of this field is quite underdeveloped.

1.2. Problem Statement

In the education system, choosing the right course and making course choices is a very challenging situation for students. It is known that the courses in the curriculum [3] are very complicated for new students at the first stage and they have frequently encountered problems. We developed the “Student Recommendation System” for the solution of questions such as whether I can succeed in this course or did I choose the correct course at the students' minds. Presenting the best recommendations to the students based on the data of the previous students will provide solutions for these problems of student course selection. It is considered that the system may be useful for showing course choices especially for the students who will start university life. The student recommendation system, considering past suggestions, also aims to solve the problems in the students' minds at the same time, while offering suggestions for courses that the user will take next semester. Besides, when students want to take course suggestion, they will be shown on the screen which courses they have chosen as a recommendation. In this section, it is more beneficial for the student to consider the recommendation and set a course accordingly. The student

will make the right choice by looking at the grades of other sample students come as a suggestion to them. In this way, they can have the foresight of which course they might be successful. In this study, we aim to design a system that guides the students for the problems in the course selection. Thus, a student who chooses a course can get an idea of his/her own choices. It is aimed to reduce the problems of course selection in this way.

1.3. Research Question

The aim of this study is a development of a recommendation system for students, which can make CBR methods and recommend a course for student course selection. During this study, we explain how we recommend the right course for the student, how we build a new system for students.

We aimed to answer the following question in this study:

- What is the success rate of the course recommendation system developed in this study?

1.4. Significance of the Study

This study will contribute to the course selection of the Computer Engineering students of Çankaya University. A student-centered approach is aimed with the system designed in this study. That is, a system is developed to assist in the selection of courses based on students' previous registrations. It is aimed to make course selections that will be beneficial to the students, and the CBR method strategy is applied for this. By using the CBR method, it is aimed to provide great simplicity in the selection of students. It is aimed at solving the problems while guiding the students and which course they will choose while making course selection at Çankaya University. In this study, we have developed the Student Recommendation System with PHP, and we aim to reduce the course selection problems of the students.

1.5. Limitations of the Study

Our system is a Student Recommendation System. We aim to develop this system with the CBR method. In this system, we are limited with the student data between the years 2010 and 2017. Another limitation of this study is that we have done our system with PHP and have completed our tests with Google Chrome browser.

1.6. Organization of the Thesis

This thesis consists of 6 chapters. Chapter 2 contains the background information related to the study. This chapter includes the history of Case–Based Reasoning (CBR) System, What is a Case Based Reasoning System?, What is a Case?, The basic steps in a CBR Algorithm: Retrieve, Reuse, Revise, Retain, and Previous Related Works. Chapter 3 presents the methodology of the study. Methodology sections have subsections; Calculating Similarity Rate of the Recommendation System and How the Similarity Calculation is Made? Chapter 4 contains System Architecture of the study. System Architecture sections have subsections; General Definition, Design Constraints and Methods – Flows Adopted in the Design, PHP – Hypertext Protocol, Structure of PHP and MVC – Model View Control. Chapter 5 presents result of the study. Finally, Chapter 6 presents conclusion and recommendations for future works.

CHAPTER 2

BACKGROUND

2.1. CBR- The History of Case-Based Reasoning System

The initial CBR systems had started to work in the 1990s when the need to logic-based problem-solving capabilities was occurring [4]. The CBR is a problem-solving system based on problems that happened in the past, in other words, it is a knowledge-based problem solving system. One of the critical features of the CBR [5] is that the solutions found in old cases that represent very similar problems are considered as the best solution of the current problem. So, the advantage for the users is that it would be saved as a case when the current problem is solved. The newly saved case can be used in the future and become applicable data for the users. In 1977, the study made by Roger Schank was considered as the foundations of Cased Based Reasoning. The system was built to record and save the solutions that occur in different situations that we are not familiar with [4]. In this context, the system was proposed as a data memorizing structure that initially was applied in situations like the medical examination and dining in a restaurant [6, 4]. Even though people did not fully understand this new creation, Rober Schank had not given up his study on the CBR system. He developed the theory of the CBR system by examining previous problems from the database. The first CBR application developed by Roger Schank at Yale University (USA) at the beginning of the 1980s, has paved the way for the further progress of CBR applications in the future. The first dynamic case-based reasoning system made by Schank is one of the oldest systems and has formed the basis of this system [7]. Also, Janet Kolodner was the person who developed the first CBR system based on the original model [4].

2.2. CBR- What is the Case-Based Reasoning System?

In general, the CBR system can suggest an appropriate solution to the current problem based on problems that had happened in the past [8]. The CBR system is necessary for previously encountered problems or experiences [9]. These experiences are data that we might previously encounter has been successfully recorded for solving the problem in the future. For example, the course recommendation system where students registered courses are recorded based upon a history-based experience, and thus the system can utilize these recorded data to make comparisons for the new students. In this way, the solution of course selecting problems can be reached for the new student. The CBR approach is designed to give course recommendations for those students who have a hard time selecting the courses. These experiences can be used to generate a solution for such problems and help the users to make the best decisions. However, at this stage, all recorded data may not be sufficient for the users. Thereby the CBR [10] system is in fact experiences collector that applies the methodology based on the experience in the past. The CBR system will provide a suggestion for each case. For instance, the CBR system will browse the past experiences for the new case; then it will be selected and compare the most similar situations. Finally, a new recommendation can be given to the users. For the CBR system to reuse this recommendation, there must be some similarities among the cases. The system would scan the database and presents the best match to the user according to similarity rates. In other words, the CBR system possesses such features that can save and reuse similar experiences in the past [11]. The outcome of the comparisons is all about predicting the possible reaction.

The CBR system is a system that finds similarities between past events and the new situation. As the results of the similarity found in the database, the appropriate recommendation will be presented. If the similarity relating to the existing problem cannot be used, the system will find the closest data in the database and present it as a suggestion. The users should internalize such a situation and act accordingly. If the data related to this situation has achieved fruitful results, it will then save the data to the database. Thus, when facing a different problem or situation, the data stored in the system will simultaneously make a response.

Moreover, because of its case-based feature, the CBR system can provide a lot of advantages [11]. For example, by storing the experiences, the CBR system can prevent errors that may occur later and avoid making the same mistakes. Nevertheless, the CBR system may not always record the data correctively. The false cases may appear due to the contingency of saving unnecessary or faulty cases. This thesis aims to generate suggestions complying with users' desire and to make decisions more effectively [12].

2.3. What Is Case?

A case is a status of recording problems that happened in the past [8]. The pre-existing problems are recorded and applied repeatedly from the database. Situations where experiences locate in are dispensable for cases [9]. In this context there are always different cases that will be shown to the users, and thus different results may occur. When using data for a particular case, it will browse the past situations for solving the new problem. As the CBR system is trying to solve the problem, it also examines the conditions, characteristics related to the problem. So, when a problem occurs, the system will analyses whether there is any similarity existing in the past problems, or whether this problem is irrelevant to past problems. This procedure will be completed by analyzing the problem at the beginning and then defining the characteristics of that problem. The situations that the CBR system examines are derived from different problems [13]. Before analyzing the case, there are some important points need to be emphasized [9]. First of all, the classification of the problem: indicating the features of the problem. And if there is a need to specify the problem, minor features should be distinguished. The purpose of this stage is specifying the problem rather than generalizing it. The second point is the diagnosis. In this step, it is necessary to understand what the problem is. Then the next step is a configuration which determines which data will be used and what items should be added and what is required. The data is distinguished and diagnosed based on this given information. In the CBR system, the case base is a collection of previously established situations [14]. There are some essential tips need to be looked at before creating a case base. These steps are expected to define the cases fully and to facilitate the process. They are as follows:

- The situational configuration of the cases
- Database model using for the cases

2.4. CBR Algorithm

The CBR algorithm is based on the reuse of solutions when similar problems reappear in front of the users [12]. In other words, most of the results generated by the system are based on similar solutions. In general, current problems or situations are the experiences users have encountered before. The CBR system calculates and maintains a similarity rate for every problem that the user has encountered before. Since data related to the past problems will be saved in the database [12]. This can be better understood by giving examples. Given the relationship between doctors and patients [15], the doctor examines the patients by receiving the complaints and condition of the patients.

The diagnosis will give the same treatment if the symptoms of this patient are similar to the formers. Otherwise, it will save as new situation in the database. In this sense, the doctor can treat new patients from the experiences of the old patients. From this perspective, the similarity system is very time-saving in our life. The doctor need not spend much time on a single patient. Consequently, it helps to increases the efficiency of taking care of every patient. The relationship of the doctor and patient is given as an example to the course suggestion system which the CBR system is used as well. The CBR system is based on finding the situation with most similar rate related to the problem occurred. It will eliminate those irrelevant results and present the most similar ones to the users. Moreover, it is where some test situation and the verifiable status are engaged in the system. In this process, the users will check the accuracy of the solution. To check the accuracy, tests are made on each data saved in the database. The tests are made to see conformability of the problem. Respectively, if the user's information or suggestions are matched with the test results, the accuracy is guaranteed [12]. The CBR system has a typical suggestion cycle. Here, the CBR system play an essential role in dealing with the problems and proposing solutions. To talk about the suggestion cycle of the CBR system, the structures involved in these cycles are 1. Retrieve 2. Reuse 3. Revise and 4. Retain [16]. For further

discussion, it is essential to look at the scenario of ‘retrieve’: when the problem is originated from the user. The most similar rate can be retained by searching all problems saved in the database and similar problems of the users. This process is known as ‘retrieve.' Secondly, in the Reuse phase, the users will look for situations with similar features. The users are reusing the information. Third, in the phase of Revise, the most similar situations of the user's problems will be presented to the user. So, it is resetting the recommended solution to the users. This structure determines what options will come up to the users in the suggestion section. Last but not least the final phase is ‘Retain,’ which refers to recording the problem in the database. At this stage, the data is briefly reviewed and then recorded. By this way, the problem can be saved as data and can be scanned and applied later following the similarity rate.

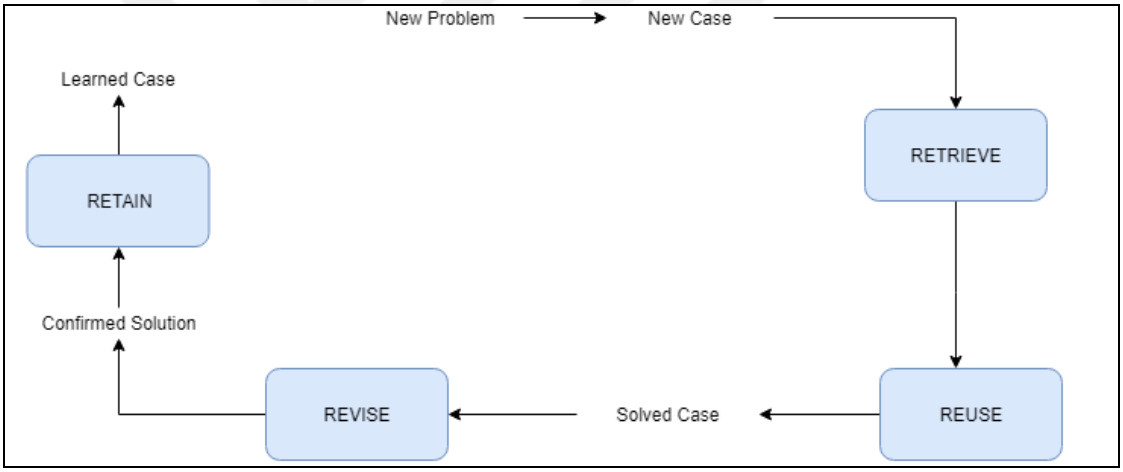

Figure 1: CBR Main Cycle

2.4.1. Retrieve

Retrieve is an essential method applied in the CBR system. It can achieve in locating the most appropriate similarities by retaking the problems that identified by the CBR system. In other words, Retrieve is helpful in pairing experiences according to the present problems. In this section, the best match will be achieved, and the process will end when the problem is found with the help of the problem situations such as the previously existing and related problems saved in the database and those resolved situations [17]. When finding the data that provides the best match, in the Retrieve

section will enumerate the similarity status in consonance with the current problem. With the assistance of the CBR system it can find the best similar solution from the database, through generating similarity coefficient among the situations based on the problem, and finally, it will list the most appropriate rate for the users. The similarity found in this phase is the closest similarity of the current problem. The method of the CBR system will examine the degree of the similarity of the cases saved in the system. Meanwhile, in the course suggestion system, there is a measurement of similarity among similar situations, and the coefficient of similarity, which sorts the result according to similar situations. In this way, a list of the most similar situations that belong to the problem could be made. This list will be presented to the user when the first three similar cases are listed in the course suggestion system. Moreover, the users can act in consonance with this list. When similarity rates are found, the quality and value relationship will be checked [9]. Each of the quality will be evaluated based on the value. In this context, in order to find the similarity rate, the following processes are required:

- If both cases are in the same state, they are given a value of 1.
- If both cases are not similar, they are given a value of 0.

These given values may differ slightly according to the setting of the systems. In such condition, the system will revise according to the problem. Also, the results coming up, in this case, will be within 0 – 1 value. The given figures between 0 and one will present the similarity rate between every two cases with the percentage.

In the course recommendation system, first, the users would decide which course to take at the beginning of the semester. The similarity rate is generated by the result of mathematical calculations based on the users' average grade and the semester information. A decent rate will be obtained according to the statistical data that multiple courses, coefficient, and the courses that students want to compare with. The obtained rate is the coefficient of users' similarity in the course recommendation system. The system will list the first three situations with the highest similarity coefficient on which the users make a superior choice depend.

2.4.2. Reuse

Reuse is the reuse process in the CBR method. The purpose of re-use is to accommodate the best matching situation to the current problem. The reuse process in the system is the solution stage for the problem [18]. It can create information of similar circumstances by adopting the historical information already saved in the system. So, it will check the situations collected in the system to see whether it is compatible with the user's situation if it is the situation will be selected. This allows the user to select similar situations compatible with their problem. Indeed, the reuse part is about the use of the case. It is a stage that by browsing the data in the past the users can reapply the data to solve their problems. It is also a situation that re-adopting new best matching solution to the problem. By locating the saved data from the database, a new solution for the new problem will be found. Finding and reusing the recorded data is the major task here. The CBR method will record this solution for the next similar situations. For any general users, the overall problem-solving process is as follows:

- What is the Problem? The definition of the problem should be neat
- Search for the most similar case for the current problem and determining the found case
- Applying the suggested solution

When in a situation with some new problems, the CBR system may not have any suggestions. Thus, some tests must be made to show how many suggestions the system can offer.

2.4.3. Revise

Revise is a process of reviewing and re-using the data in the system. It includes evaluating information about the current situation and obtaining the required information. If the result is successful, the system will record this situation, and thus the successful situation is learned [17]. It is also the stage to check if the successful situation is applicable or not. The Retrieve stage contains two different types. They are:

- Working with a specific database
- The unfix database is used only for search work [17]

The course recommendation system work with a fixed database by the retrieved algorithm. Retrieve should be applied in this situation and the recording phase should follow this path. The problem itself and its content should be examined within the situation (case). The relations between the solution and causation of the problem should be considered. If the result of the solution of the relation or the event is proved correct, the revision stage is operating soundly. This will be saved as part of the reused material for the future, and the results addressed systematically will proceed for the new problems. In this case, a suggestion is offered to the users. However, at times, users' feedback could be positive or negative. Both of the conditions occupy an essential place in the CBR system. It is because just as the successes achieved by the suggestion system, failures should be presented as well.

2.4.4. Retain

The Retain stage in the system can be beneficial to offer similar solutions to future problems. Under this procedure, the solution in the system will be saved as successful to the database. When facing a similar solution, the database will be checked, and the most similarity suggestion will be presented to the user. Also, if the problem is not previously recorded, it will be saved to the database [17]. If the situation is saved, it can be used for updating.

2.5. Previous Related Works

In this chapter review of related literature will be presented. In the literature, different studies employ Case-Based Reasoning (CBR) in different subjects. For example, in their study, titled "A case-based reasoning approach to cognitive map-driven tacit knowledge management," Noh et al. work on the subject of credit analysis [19]. Techniques like CM (Cognitive map) and CBR (Cased based Reasoning) are used for saving and re-using personal data effectively. In this article, CM is used to formalize the information [19]. The CBR approach has been used to get more precise results. The methodologies used in this research include three

stages. First, CM which is used to define the information; Second, CBR is used for adapting the information; and for the last, CM which was applied in the previous stage would be used again to make a sound conclusion. To ensure the accuracy of these applications, a sample of credit analysis has been tested. The results show that the given cases have been applied to the problems and the appropriate solutions are produced. The given recommendations have been observed to be appropriate with the test results.

The aim of this article is about the application of CBR system in different fields. For instance, in the "Building CBR systems with Jcolibri Science of Computer Programming," an application tool developed by Java is used [20]. The aim of establishing the Jcolibri application is to create the CBR system based on the object-oriented framework. This study will discuss JCOLIBRI, an object-oriented framework in Java for constructing the CBR systems, that mostly takes advantages from reusing formerly developed CBR systems.

CBR is considerably useful technology in tackling problems that are categorized as a subfield of artificial intelligence. The CBR system is thus significant in conducting any application and determining methodologies. The methodologies used in this article were mostly finding similarities and implementing codes into the project in the field of reuse. There are two essential factors: first, it is the importance and applicability of the other systems that are created through a CBR system. Second, it is the possibility to take advantages of real-life experiences and use these experiences in the CBR system. This system has been tested for its accuracy and capability. The result shows that this system can ease the problem-solving procedure. The goal is developing the application with the contribution of the users' overtime and providing the handy reference for the users by the CBR system [20].

The following article will explain the use of the CBR system in a different area by looking up the thesis of "Information extraction from case law and retrieval of prior cases." It shows the previously made court decision and applies a database for using the previous cases related to the decision [21]. This research's purpose is to access documents that may affect new cases by a combination of information acquisition

and machine learning techniques [21]. Machine learning techniques are used for creating recommendations for current problems [21]. Thus, this provides great benefits for this research. This system is known as “History Assistant.” Besides, the general purpose of this system is to match the practical applications of this system with the knowledge from the literature. This system provides a test. In the testing phase, the performance of people on the system will be checked. 60-80% of the accuracy of the result can be ensured. The final part of the study will mention in which ways the system solves various specific problems. Established algorithms and the data are used for specifying the system and avoiding generalization when presenting a recommendation.

The other focus of this research is aimed to develop information-based systems. In “Knowledge engineering issues in developing a case-based reasoning application” the model of human memory in cognitive psychology related to the CBR system was studied. Besides, the memory storage is examined [7]. This study will deal with this situation in two different ways. First of all, it is analyzing the current problem; the second is to see the effects of the CBR system on the case mentioned in the first method. In this research, an iterative methodology is used in the field of giving a recommendation. ISAC, used for these assessments, is the name of the system. This system is designed to be the solution to prevent aircraft crash with the help of the CBR system. The methodology used in this research is obtaining error analysis by the work of experts when creating the recommendation and thus providing the new recommendation as a solution. Some tests have been conducted to strengthen this thesis.; a “LeaveOneOut” strategy is implemented in 50 different cases during the tests [7]. As a result, a very successful rate- 95% has been achieved in the recommendation system. Nevertheless, it is not satisfying, because this study aims to develop this system further.

The next study is named “Case-based Reasoning for Web Service Discovery and Selection.” [22]. Here, a recommendation will be presented by using the CBR system. The goal is to develop “Interface Compatibility Scheme” [22] with the help of the CBR system. Also, a data set consisting of 62 Web services will verify the application and then make recommendations for approximately 90% of the case by

the CBR system. As a result, some essential facts may appear in the test application. In this study, this application was tested in three different ways. The test result is generally congruent with the performances of the system. It can receive feedback from service servers, and useful data is acquired in metric. This study is designed to advance and develop for the future.

The next research is about "Mining content case bases for case-based Reasoning" [23]. In this study, there are different usages of the CBR system. To solve these problems, we used our algorithm, Kernel-based Greedy Case-BaseMining (KGCM) [23]. The tests used in this study were evaluated separately for each case. "CNN-FP, COV-FP, RFC-FP, RC-FP and ICF" algorithms have been used in adding case and case extraction to the system.

The other study is "Using background knowledge in case-based legal reasoning: A computational model and an intelligent learning environment" [24]. The CBR system is widely adopted by lawyers. It is a system used by lawyers to consult the past case data for searching for a solution to the problem. The methodology applied in this study is "Issue-Based Arguments" [24]. This methodology is used to organize the data related to the subject. "McNemar" test was applied to test the accuracy of the "CATO" model in this study. This test is made for checking if the observations are different from the test result. According to the result, the CATO application has an advantage which is a theoretical contribution to this research.

The next article is "Loss and gain functions for CBR retrieval" [6]. The working of the CBR system is based on the similarity of the cases. Similarity plays a significant role in determining the similarities of the cases. Two different factors were discussed in this study. First is the similarity. The other is the degree of similarity among cases. In this research, the methodology applied also reflects the utilization of the CBR system. With the help of the CBR system in making a new decision, methodologies are adapted to present the appropriate recommendations. Moreover, the test is made by this methodology. The tests and methods carried out in this research are prepared for particular problems. What is more, the test is also made to examine the

performance of the application. Correspondingly, the test result has achieved in providing the best recommendations for the users.

The following article is “Using case-based reasoning for the design of controls for internet-based information systems” [25]. The CBR system will use the similar case with those of the IIS monitor. It also suggests the most convenient IIS monitor in the system. In this study, data mining techniques are studied to prevent network attacks when finding a solution to the problem. Hence, the use of the CBR system provides accurate historical information of the IIS monitors and resolves problems.

The following article is “The search for knowledge, contexts, and Case-Based Reasoning” [26]. The process used in this research is to analyze the existing problems in pairs. The problems are examined in pairs to give a recommendation. In this part, the saved data set will be checked. As a result, the given cases have been applied to the problems and the appropriate solutions are produced.

The last article is “Case-based reasoning is a methodology, not a technology” [27]. This article is questioned about if the CBR system is an Artificial intelligence (AI).

As to methodology, it is essential to see if the methodology used can provide better suggestions for solving problems. There are some procedures when implementing the CBR system in solving a problem. This research is examined for the solutions to the problem. Consequently, the systems mentioned here are using technologies in different fields. However, there are always the same procedures that need to be followed when generating a solution.

CHAPTER 3

METHODOLOGY

Many applications are using the CBR system found in the result of the literature review. However, there is no similar study relate to the problems that students have in the course selection. We developed a web application that suggested courses from those who have similar courses during the course selections. This system is operating based on the WAMP server. The large-scale database connected to the WAMP server where students' data are saved and using the relative data through the CBR method can create a new suggestion system. The tasks of the users are taken as the main functions of the system. After completing the database, the structure based on students' data have made database very important on setting up the system. This system is designed as user-friendly so that students can understand the system quickly (see Figure 2). The system consists of three parts as follows:

1- Transcript page that shows the students' previously taken courses
2- The curriculum page that presents the students' next semester courses
3- The course information and recommendation page that shows the first three most similar students' courses determined by using CBR method used in this study

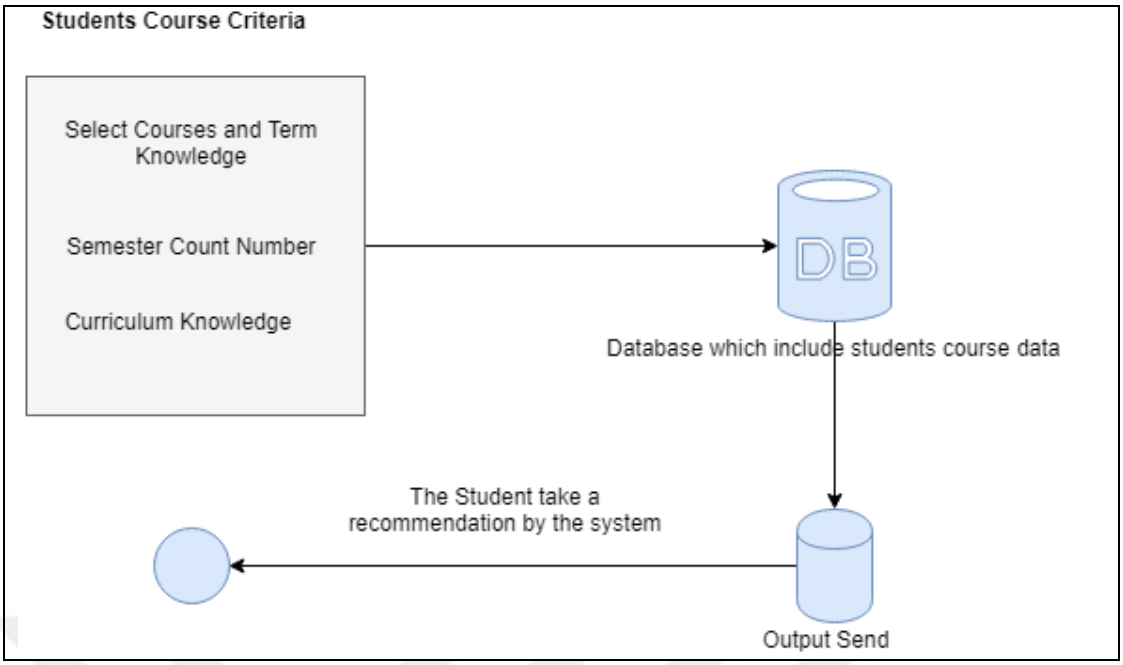

Figure 2: CBR Cycle for Student Select Course

As shown in Figure 3, the first page is the login screen that will be redirected when the user wants to log in to this system. There are also two language options for the users by the add-on code.

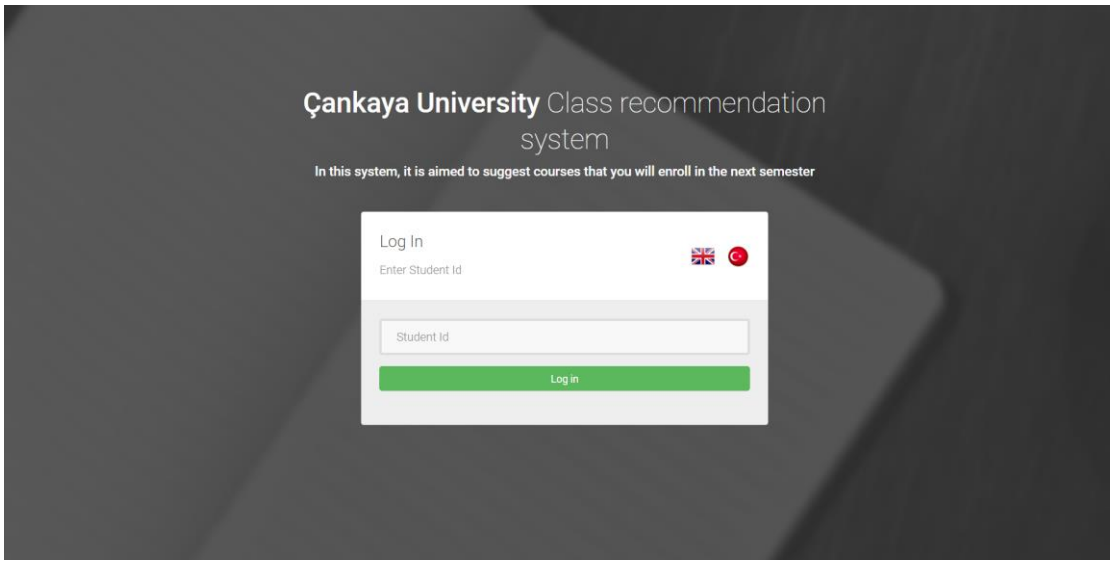

Figure 3: System Login Page

When the user has logged in to the course recommendation site, the first page to be opened will be its transcript screen (see Figure 4). Students' information that is saved

in the database can be retrieved. Course information such as course code and credit information will be displayed on the student's transcript screen.

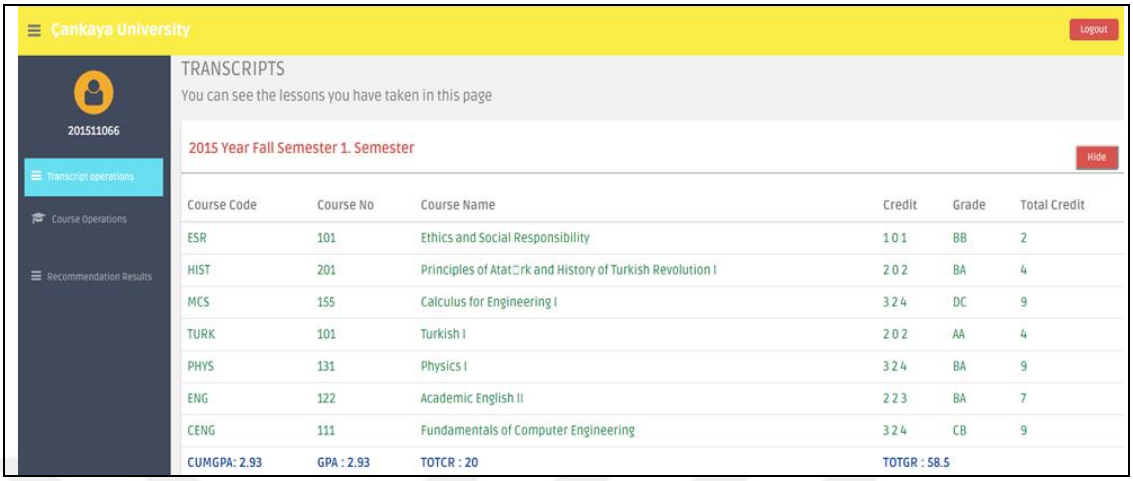

Figure 4: User Transcript Page

Also, we provide the user with information about the courses load, such as CUMGPA, GPA, and Total credit. This page lists all the lessons that the user has taken so far. If the user failed the course, after system check this filed will be shown in red. Courses are shown in green as the user passed the courses received. For the convenience of the users, the calculated GPA information is also provided. If the user wants to exit this page, he/she will be redirected to the login page again.

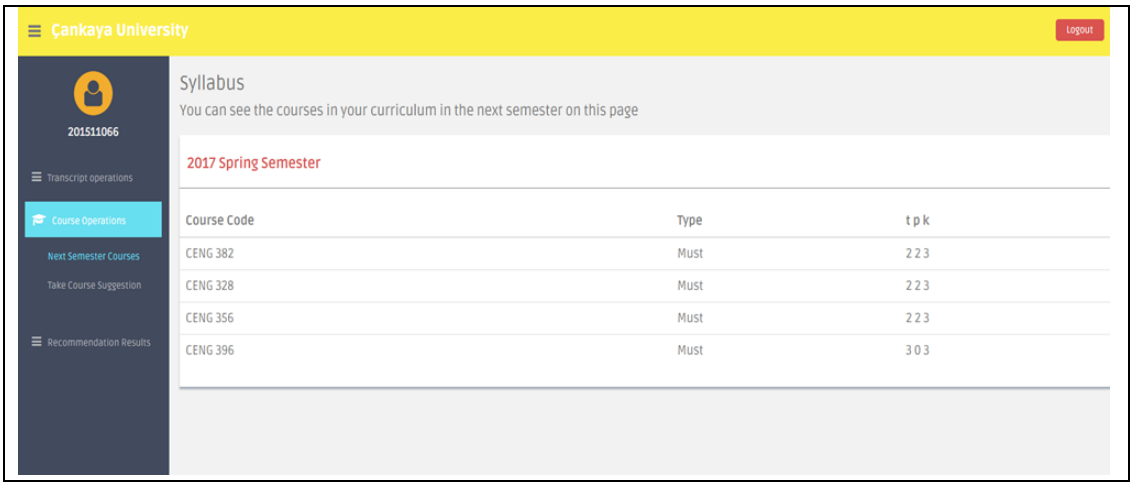

Figure 5: User Check the Syllabus

Figure 5 shows the Syllabus (curriculum) page where the user will be informed about the courses that should be taken for the next semester. This page contains all the

course information that belongs to the user. According to the curriculum information published on the "Ceng.cankaya.edu.tr" website, the opened courses are recorded in the database. The syllabus page will list the lessons that will be opened to the user after the current semester by finding the recent semester information of the user. The following steps are made to list the information:

- The registered year of the user is identified.
- The recent semester of the user is located.
- The next semester of the user is determined.
- The courses opened for the next semester are withdrawn from the database and shown on the page.

As shown in Figure 5, the list of courses that the user must take is shown. Depending on the user's class the system will show must courses in the following semester. This curriculum information is facilitation to help the user making the decision when selecting courses, even in the condition of without recommendation or after receiving the recommendation.

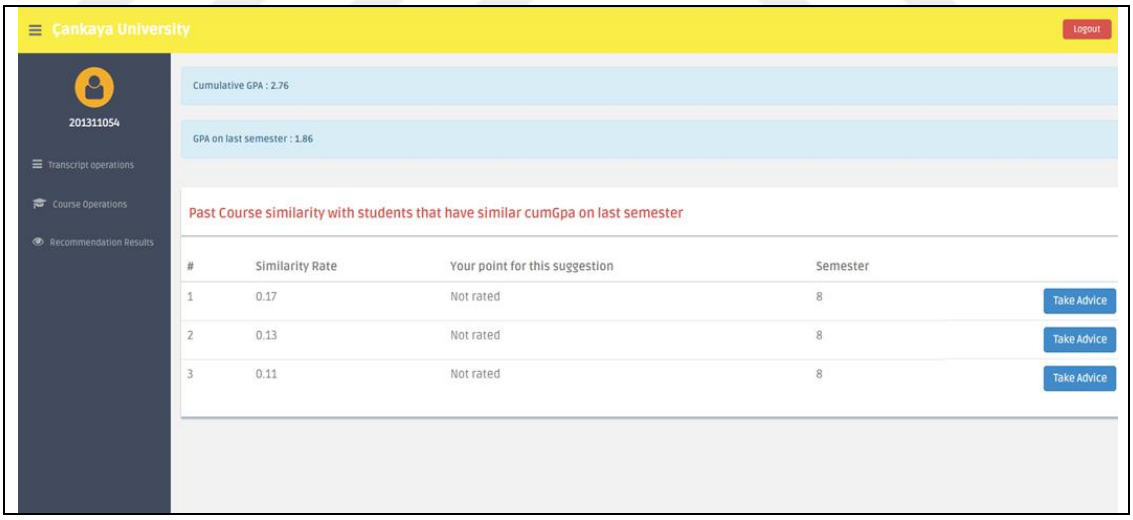

Figure 6: User Take a Recommendation

According to Figure 6, the user should click "Take Course Suggestion" when he or she wants to get a recommendation for the lessons. Three people with the most similar will be listed on the page, based on the suggestion algorithm defined for the user. The screen lists some of the user's information. They are:

- Cumulative GPA information
- GPA in the last Semester information

This page is obtained by comparing the information of the user and other users who might be the recommendation. When making this comparison, it is essential to calculate the similarity rate of each similar user and the user who has logged into the system. People who are most likely to resemble the user will be listed according to decreasing similarity rate.

The similarity rates are calculated according to specific metrics, and thus the results are based on these metrics. To find the similarity rate an evaluation method was made. This rate can be explained as follows:

All users registered in the database with the user to login to the system must be compared with their course information. Some calculations were performed during the comparison process. The course codes and grade of the courses taken by the user registered in the system and the selected users will be compared according to the criteria specified in the database.

The comparison process is carried out according to different semesters and the courses opened in the semester. This comparison process continues as long as the user is logging in the system. Then, we able to get the result of the similarity rate by using the metric calculations. With the result of the similarity rate, the score will be listed decreasingly. Therefore, the three people who are closest to the current user will be selected and shown on the screen. Similar users listed on the screen will be recorded in the database. For making a future suggestion, the new user is recommended by looking at the individual who recently dedicates to record. Which users are compared with whom will be saved in the database. The calculation example used during the comparison is as follows:

Table 1: System Calculation Example

Student A	Note	Student B	Note	Coefficient	
Course1	FF	**Course1**	FD	1	Two people fail the course
Course2	BB	**Course2**	CB	0,83	Two people pass the course
Course3	FD	**Course3**	CC	0	One of them pas the course and one of them fail the course.
Course4	BB	**Course5**	FD	-1	If the course is not equal.
		Course6	AA	-1	If the course is not equal.
			Recommendation Point:	(0,03)	All coefficient total point.

There are three different situational criteria to calculate the similarity rate. The criteria are as follows:

- The number of the semester should be determined
  - The semester in which the user has attended should be shown
  - The numbers of the semester of the selected students to be compared should be controlled
- Taken courses must be sorted (comparison between semesters should be made)
  - For the user to receive suggestions, details of the semester should be looked during the comparison.
  - Each semester should be compared on its own
- Curriculum information
  - Other users in the system and the users logged in the system should register in the following semester (students who get suggestion should have enrolled in the courses earlier.)
    - Students who get suggestion should enroll in the current semester and have attended the past semesters.
- The students selected to provide suggestion should enroll the university earlier than the current users

3.1. Calculating Similarity Rate for a Case

When the course suggesting the system is making assignments for the cases, it must require cluster function. Each object must have a U cluster [9]. Each case belonging to the cluster must have a corresponding set of values that it can be found and matched. Moreover, we need to acquire the condition shown in Formula (3.1).

$\theta \in A$ (3.1)

The purpose is to examine the current problem and then try to find a situation that can solve this problem. When addressing and evaluating the problem, these cases will help in understanding the condition well. The case which is most similar to the problem should be identified and taken into consideration carefully. This will help in generating plausible suggestion. The solution to each problem should be similar to the cases because the problem is solved by consulting similar cases. In this sense, it is necessary to review the similarity status by using the attribute-value [9] relationship. When assessing a problem, the case bases are as follows (see Figure 8):

- Examining the relationship of the similarity between the attributes
- The relevance of each similar case

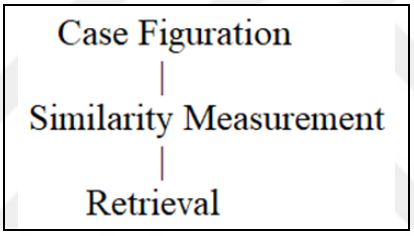

Figure 7: Similarity Process

The similarity process is initiated by these two situations. This is one of the best ways to make a comparison with the existing cases. According to this situation, if the case is chosen to measure the similarity,

- If the cases are similar, the value of 1 will be given
- If the cases are different, other value will be given

Objects are checked to identify the similarity status. While comparing these objects, the underlying set process is shown in Formula (3.2) and (3.3).

$S(a,b)$ (3.2)

- If two cases are similar, it depicts that "a and b are similar."
- If two cases are different, it depicts that "a and b are different."

$DS(a,b)$ (3.3)

This method was also used in the "Course suggestion system" to identify the similarity situation. In order to provide the most accurate and efficient measurement of similarity to the user, the metric calculations were made in this system.

As Figure 6 shows the process of finding similarities in the system, and it is described as follows:

Calculation of student A and B's similarity rate:

- Student A and student B both failed in the same course: '1' is given.
- Student A with student B both passed in the same course: 'average' is given.
- The similarity rate of those letter grades is calculated in 4-grade point.
- Formula: 1/(high note weight/low note weight)
- If one of the students pass and the other fails: 0 is given.
- If the lessons taken are different between the students: -1 is given

When calculating the similarity rate, the values obtained from above are collected and divided by the number of different courses taken. So, all the collected coefficients are divided by the total coefficient. So, there will come up with an average which represents the similarity rate between the students. In this way, students' courses are continually compared on the basis of the semester. This process is repeated separately for each semester. As a result, the most similar three people are selected and listed with regards to their similarity rate for the student who has entered the system.

Table 2: All Notes of the University Student's

AA	4.0
BA	3.5
BB	3.0
CB	2.5
CC	2.0
DC	1.5
DD	1.0
FF	0

Similarity Assessment: CBR is used to find similar operations, sorting and pairing the appropriate data. The pairing process is determined by the total number of indexes. During the pairing process, CBR scans the data in the entire database and searches for the data similar to the problem and pairs the data that it finds [28]. The algorithm used for finding the closest data has been stated and calculated in the following formulas. The following formulas are used to find the closest data in the CBR system and to pair the information that is found. For the calculation of similarity, the numeric function for the N property is as shown in Formula (3.4).

$$S_n = \frac{W_1 \times slm\left(f_1^{input}, f_1^{stored}\right) + W_2 \times slm\left(f_2^{input}, f_2^{stored}\right) + \ldots + W_n \times slm\left(f_n^{input}, f_n^{stored}\right)}{W_1 + W_2 + \ldots + W_n} \quad (3.4)$$

The given total values will appear between 0 and 1. The number between the resulting 0 and one is calculated as follows (see formula (3.4)).

$$S = [0,1] \; or \; S(\%) = [0,1] \times 100 \quad (3.5)$$

If S = 1 two cases are equal if S = 0 two cases are entirely distinct [28]. This formula will be used in the student recommendation system to calculate the similarity coefficient as shown in Formula (3.5) The similarity coefficient is calculated according to its different properties in the formula.

Similarity function determines the similarity rate between the current problem and the solved problems in the database [28]. That is, it performs some calculations for finding the similarity rate by using numeric values. After these numerical values are calculated, the system will play an active role in distinguishing the similarity rate of the situation and lists the problem regarding the similarity rate. In this way, we, users

can understand the similarity the rate of the existing situations. In order to find these similarity calculations, these formulas use different numerical analysis techniques, such as logarithmic calculations, logic operations, linear or exponential functions, and artificial neural networks [29].

In the CBR method, it is essential that in which ways the similarity rate will be found. In this study, the nearest neighbor algorithm was used to find the similarity rates as shown in Formula (3.6) [28].

$$S = \frac{\sum_{i=1}^{n} w_i \times sim\left(f_i^I, f_i^R\right)}{\sum_{i=1}^{n} w_i} \tag{3.6}$$

This nearest neighbor algorithm, which is calculated by the above formula, is used to find out which course results are the most similar to the students. So, the most crucial point here is the sum of the similarities of each feature. The other cases with the same numerical value specified in this algorithm are only used as a basis. In the system, the top three similar students will be listed. Only the listed ones can be taken as cases.

3.2. How the Similarity Calculation is Made?

The user who has logged into the system will be directed to the new page by clicking on the "Get Suggestions" button if the user wants to receive suggestions.

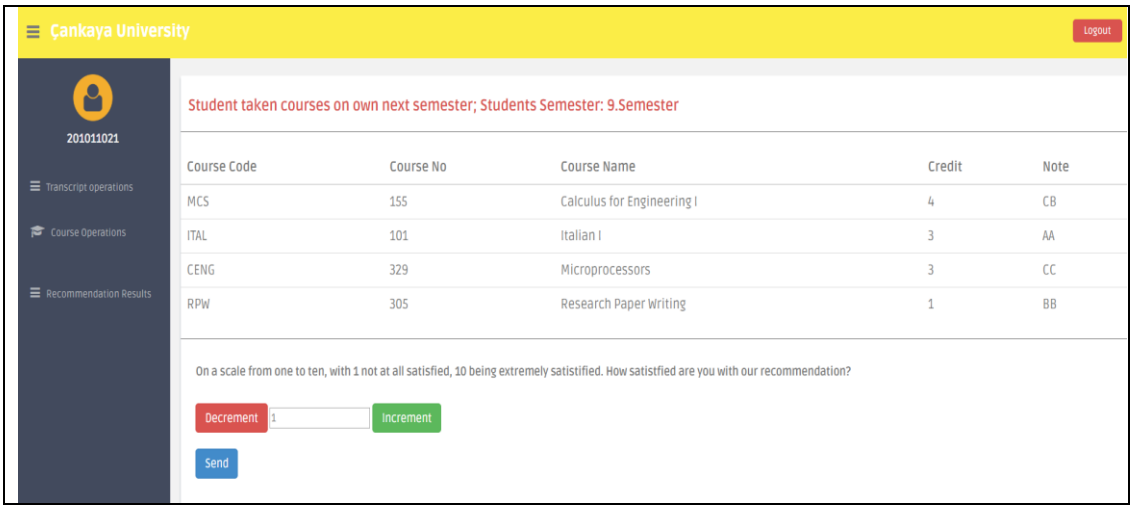

Figure 8: User's Recommendation Page

As seen in Figure 14, the system will present the most similar candidates, and the courses they will take in the next semester will list on the screen. Therefore, the user can see which courses were taken by the most similar candidates for the following semester. Besides, not only the courses but also the course grade of the comparing candidates will be shown on the screen.

The title from above will give the information about the suggestion list of each semester for the users' convenience. In this section, the user is also offered the possibility to evaluate given recommendations. The evaluations are critical regarding the system's testability. Moreover, the user's satisfaction is an essential basis for the system. The users can scale the given suggestions by the compatibility of course information. The scale ranges from 1 to 10; one means 'I am not satisfied', ten means 'very satisfied.' The evaluation section is meant to provide feedback on how satisfied the users are with the recommendation. The results of the evaluation will be recorded in the database so that it will be easier to measure the satisfaction rates of users.

The pages that are displayed in Figure 15 and Figure 16, are created, and designed for testing the course recommendation system. The test has been made with students, and it offers detail about the recommendation system and the taken courses. The course recommendation system is designed to evaluate the percentage of the success rate.

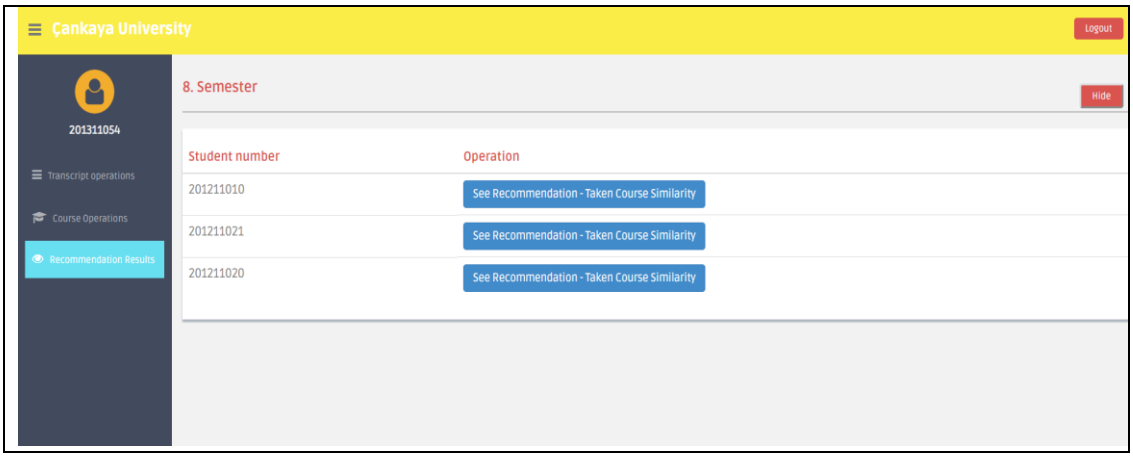

Figure 9: User's Recommendation Page 2

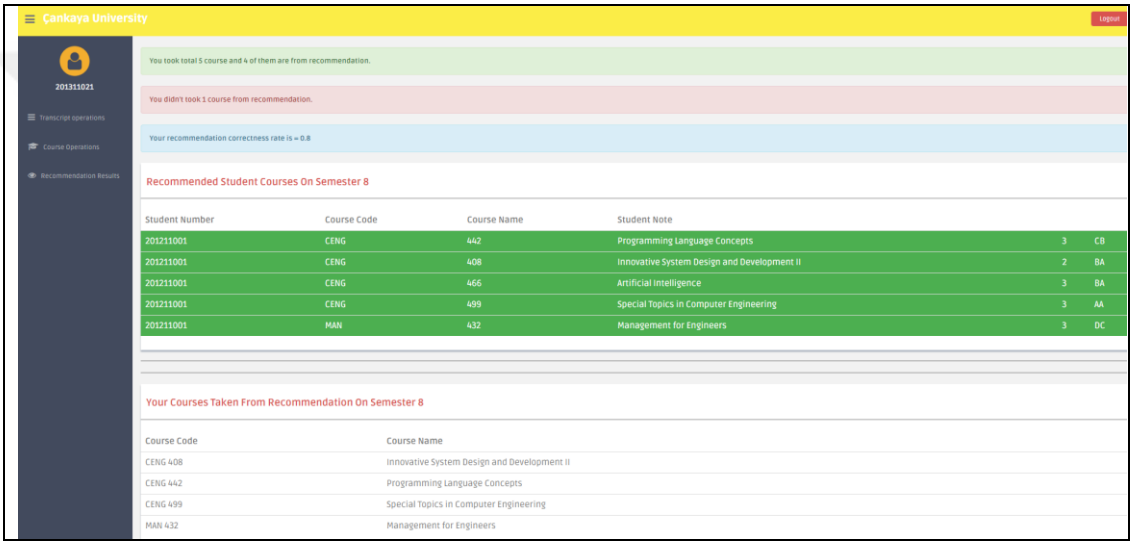

Figure 10: User's Test Screen 1

CHAPTER 4

SYSTEM ARCHITECTURE

4.1. General Definition

The system architecture describes the relationship between the applied technology including structures and modules created for the course suggestion system, the necessary components, and structure of the system and the technology. The approaches and methods that were implemented in this project will be discussed. The Cased Based Reasoning (CBR) approach used in the course recommendation system, will be discussed in this project as well. The system architecture is about processing the general functioning of the system, and the condition of its sub-relations, the system's functioning diagram and the schema of the CBR approach. The infrastructure or say framework used in the course suggestion system will be discussed. Besides, the functioning of the subsystem of course suggestion system including the structure of the filing, architecture (MVC) structure will be examined.

4.2. Design Constraints and Methods- Flows Adopted in the Design

PHP language is used in implementing the application. By writing the PHP language, the YII framework application becomes easy to create. The purpose of using the PHP language is because it is an open source code and it is convenient for Web applications. When doing the project, the construction of interface is created by HTML pages, and their connections are provided by code. Sublime Text Build 3143, WAMP Server 3.1.0 and phpMyAdmin was used to perform the application. MVC was used as a software architecture model in the project for the course suggestion system. Besides, object-oriented methods are used during the design and coding process. The overall functioning of the system architecture is explained by the flow diagram.

4.3. PHP – Hypertext Protocol

Rasmus Lerdof first introduced the first PHP language in 1990 [30]. The first purpose of Rasmus is to publish on the Internet environment. Rasmus began and kept developing the PHP language which is opening of the Staff Homepage. Thereafter, this language began to draw much attention, and the form structures on the Web pages were created. The structure that enables the processing of information through the form was added to this structure, and it is named PHP [14]. In 1995, Rasmus established a team to develop this language and made it object-oriented. Also, this language is designed to resemble the C language for those who want to develop similar languages.

Although the PHP language has not been developed for a long time, it is still being used and becoming famous for web development. Nowadays, the PHP 7.2 language, the latest version and was released in July 2004 is currently applied. It Contains a new object model and a new feature called Zend Engine 2.0.

When the web page was written by PHP language, a scripting language is connected to a browser:

- A PHP file is called by the browser and the client
- The server will understand that this request is a PHP file from the extending file and send this to the PHP interpreter. The PHP interpreter translates the language into machine language.
- The PHP interpreter also executes the code in the file and sends the result to the server.
- The result from the server will be interpreted into an HTML page to the clients.

4.3.1. Structure of PHP

PHP is an open source code. Generally, there is a structure used to enhance the web and embedded in HTML. PHP codes have a unique start and final status. A PHP code is specified as <??start with PHP and? > and ends with this statement. PHP codes can only be written between the tags that have previously appeared.

These PHP tags can be easily opened, and the code can be written in it. PHP codes can also be embedded in an HTML page. It would be advantageous to open the HTML code in PHP tags and write the required PHP code instead of writing lines of HTML codes. The PHP is a language that can also be interpreted on the server (HTTP). Because of this feature of PHP, dynamic web pages and the connection to the database can be created. In order to make a database connection with PHP, it is necessary to install the database application. The MySQL database is where the PHP language operates. Once similar database modules are installed for the PHP language, the database can be connected [31].

Also, the PHP language can work on all major operating systems such as Linux, Windows, Mac OSX, and Unix variants. When writing PHP code, the writer needs to decide whether the code should be a functional or object-oriented approach.

```
427     public static function findLastSemester($student_number)
428     {
429
430         $criteria = new CDbCriteria();
431         $criteria->select = "MAX(semester) as lastsemester, MAX(year) as lastyear";
432         $criteria->condition = "student_number = :snum";
433         $criteria->params = array(
434             'snum' => $student_number,
435         );
436         $model = StudentsGrades::model()->find($criteria);
437         return $model;
438
439     }
```

Figure 11: A Basic Code of the Project

As seen in Figure 17, there is an example of an object-oriented PHP code. By the aim of locating students' last semester, PHP code which was used in small function YII framework is written.

In compile time of PHP language coding, the source code compiler can make compiling easier. If the code side is encountered with an error, the source code will make corrections after compiling the code. Although this process is sufficient to fix an error, there are some disadvantages. The source code is re-interpreted whenever the program runs. Because of this, it will lose speed compared with other languages.

In order to compile a code written in PHP language, the PHP interpreter must be installed on the computers [30].

4.4. MVC – Model View Control

MVC, an abbreviation of Model-View-Controller, is an architectural pattern used in software engineering. It is based on the arranging and representing a large amount of data and in many complex applications. The MVC design pattern is advantageous in many areas. The MVC design in Web applications is useful in shaping complex projects [32]. The coding phase in this project is not a simple task. If there is no proper separation with the MVC format which means that there is no error in the code side, the programmer must deal with this problem for hours. This can also lead to errors in coding and design or changes in code structure [33]. In this part, the MVC can provide facility if there are any changes happen in the future. This means that any modification of the MVC design needs to be done on the side of the project code or interface [34]; and by distinguishing the structure that serves a different purpose of the project, the testing and further development of the code can be achieved. MVC is aiming at giving facility to the users. This facility is based on a three-way divided application which provides usability of the project and convenience to the users. These three parts include Model, View, and Control. The model part is the part that represents the essential part of the project. The View section is the part where the activity of the created model and the user or customer take place. The controller is about the interaction between model and view [35]. In an MVC project, this request is forwarded to the Controller section when the user requests the View section. The Controller contacts the model to bring the desired data and reaches and calls the class structure within the section. The results will be shared by the View section, and thus feedback will be provided to the user. These sections are explained below:

Model: This section is the part where the objects that will be used in the project are created. It also represents the data that will be used in the application. It Is where the logic of processing data is made. Generally speaking, it is where the process of

registering data and retrieving the data from the registered location and the same tasks are performed.

View: When the project is completed, the user will see the interface part. According to the project structure, this design can be created in different ways such as web page, desktop application, or mobile design.

Controller: It provides the communicates between the Model and View. It transfers the received data from View to the Model section. The data obtained from the Model section will display in the View section to the user.

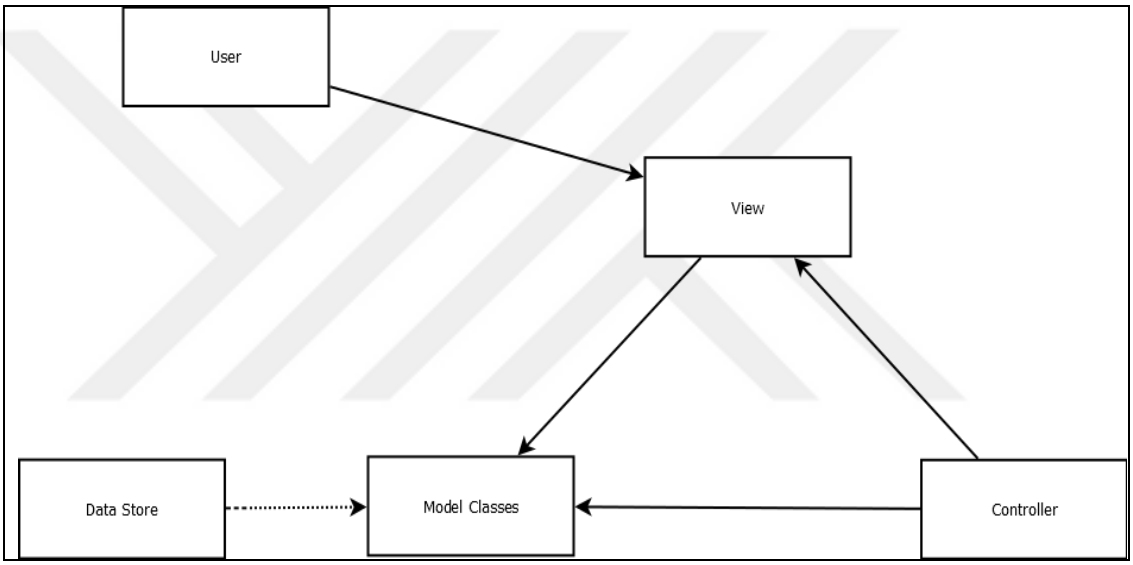

Figure 12: Depicts the Structure of MVC

CHAPTER 5

RESULTS

There are some criteria at the phase of testing in this study. Students studying at Cankaya University were chosen in this test. Database of these students was revised, and thus the updated information will be recorded. In this case, we can acquire the courses information of the 12 students each semester. The information is kept in a different table. With this information, it is possible to know the degree of success of the course suggestion system. Also, this page was also made for the test to see the success percentage of the project. This page is only active for these tester students which means other users cannot see the details of this page.

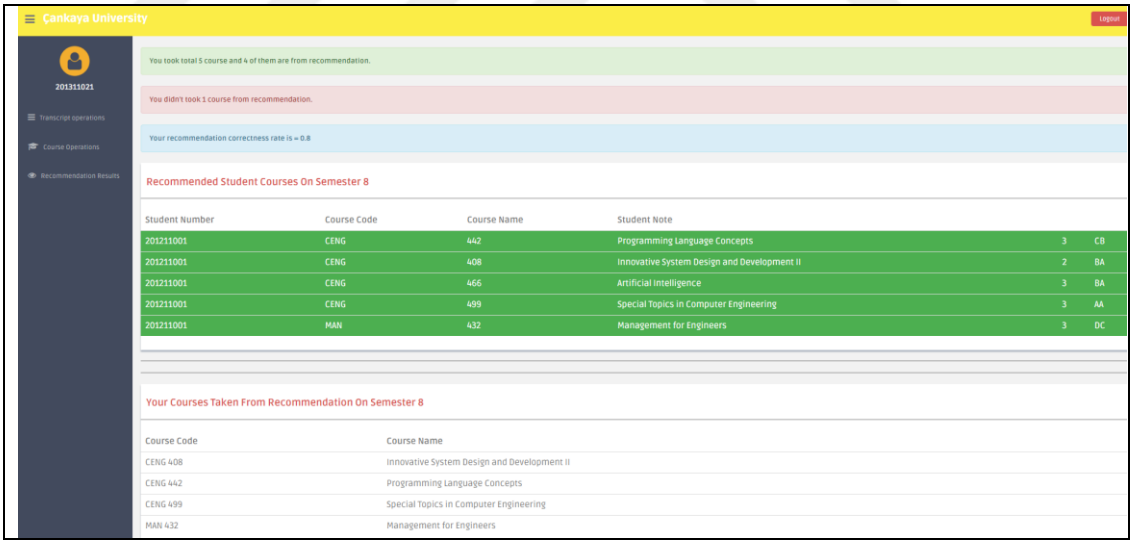

Figure 13: User's Test Screen 2

As shown in Figure 19, this will be the next screen when the tester student clicks on the "Recommendation Results" section. The test carrying on in this page is based on the single semester. The courses information in the last semester of the tester students were recorded in the database. Afterward, the test is completed by comparing the courses the user has taken with the suggestion system offer. The most similar three

students with the user presenting by suggestion system will be listed again. When the user presses the "See Recommendation" button, the screen appears as shown in Figure 19. When the user clicks on the most similar candidate, the user can obtain relative information about the candidate. As shown in Figure 19, the following information is shown on the screen for the user;

- The course suggestion system has offered six different course proposals for the user. The one of the information from the suggestion system will appear in green color on the screen.
- If the user is in the red area, the information about the courses that have not been taken will be shown on the screen.
- The last information is provided through the test. Therefore, with the suggestion system, a ratio of taken courses is shown on the screen.
- The course list appear in green are is provided by suggestion system for the user to select appropriate courses.
- At the bottom of the page, it will show the current course information of the user.

The overall algorithm of our recommendation system is presented in Figure 20.

1. *Start*
2. *Look at the number of the students who log in the system*
3. *Find the student number of the user logged into the system*
4. *Check all data which are smaller than the user's student number*
5. *Check the user semester number and find the semester number.*
6. *Determined the semester (database definitions: fall – 1,2,7, spring – 3,4,8)*
7. *Compare current student's courses with other students' courses*
8. *Find the similarity ratio*
9. *List the student's similarity ratio from highest to lowest*
10. *Choose the first three people which similarity ration is highest*
11. *End*

Figure 14: Design Algorithm of the System

These test results of the selected ten students are presented in Table 3 below, and their average is calculated. There several essential steps while calculating the average of the students. They are,

- The similarity rates of each tested student are calculated automatically by the system and are displayed on the screen.
- For each tested student, the system will present three different recommendations (students), and their different similarity rates will be listed on the screen.

Table 3: 10 Students' Recommendation Results

Test Number	Student Number	Advice 1 similarity	Advice 2 similarity	Advice 3 similarity	Total similarity rate
Student1	201311021	0.8	0.6	0.6	2
Student2	201311022	0.25	0.5	0.38	1.13
Student3	201311023	0.75	1	0.75	2.5
Student4	201311026	0.67	0.5	0.83	2
Student5	201311027	0.5	0.5	0.5	1.5
Student6	201311030	0.8	0.6	0.8	2.2
Student7	201311032	0.6	0.8	0.8	2.2
Student8	201311037	0.5	0.67	0.5	1.67
Student9	201411070	0.57	0.43	0.57	1.57
Student10	201411036	0.6	0.8	0.6	2
					Average: 18.77/30 = 0.62

In order to make the calculation more manageable and faster, the recommendations in the table presenting to the test students are in decimal format. For example, test student 1 with the number of 201311021, his/her recommendations will be noted as 0.8, and 0.6. In other words, these similarity rates format in percentages. The similar rate of 80% indicates the similarity between the taken courses by the 1 test student and the recommendation proposed by the Student Recommendation System. In the same vein, the second suggestion would present 60% similar rate. While calculating the average for the test students, there might be three different possibilities occur in each tested student. Therefore, dividing all the suggestions we have gathered from the ten test students in the system by thirty will lead us to the right conclusion. As a result, the total score we found is 18.77. Since each student has three different possibilities, we divide 18.77 by 30; then we can obtain the number 0.62 which indicate the right recommendation average rate as 62%.

Table 3 also shows the detail of the system's recommendation for the test Student 1 and the courses taken by this student in the semester.

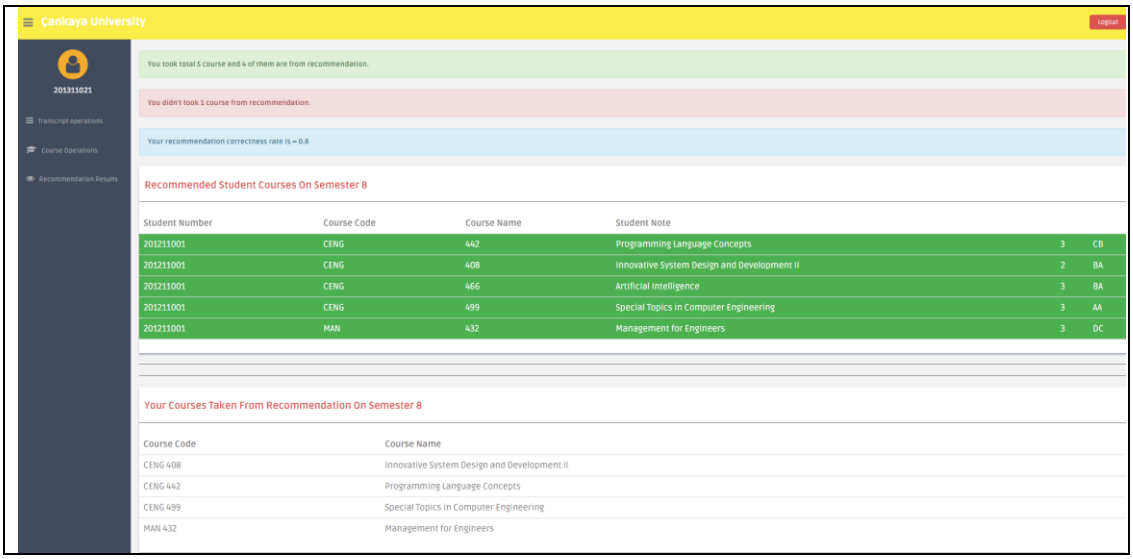

Figure 15: User Recommendation Result Page

The page shown in Figure 21 belongs to the first test student. When the student clicks on the first suggested student in "Recommendation Results" section, he/ she will reach a recommendation result page. As seen in Figure 21, the suggested student-generated by the system has taken five courses in total. Also, the Test student took 4 of the suggested courses, only one of the suggested courses is not taken. The screen lists which courses are taken by the student in the colorless section. The student who is presented as a recommendation to the test student is shown on the page in green color. By the color differentiation, the test student can easily understand the courses that are not taken. Besides, the course selection similarity rate calculated automatically by the system is shown above. This person's similarity rate is shown as 0.8. The average of the test students was calculated by looking at their similarity rates displayed on the pages. Then, we can obtain the accurate recommendation rate of 69% from the recommendation system. As a result, Various tests were performed to ensure the test of the system is successful. The test of the system has been made with multiple users and giving the correct result has been tested. The system was tested with 100 students' data, and the total averages of the recommendation correctness rates firstly come up with highest recommend student seen on the screen by these test students, and the generated result shows the success of the system. As a result of these calculations, a success rate of 69 percent was achieved. Another test criterion, the success percentage of our system was determined by taking the total averages of the recommendation correctness rates

taken by the test student for the three recommendation students in Figure 21. According to the data of these three recommendation students, the system has achieved success with a ratio of 69%. According to the results of each proposal, the highest test success rate was achieved by 72%. The system has achieved successful results in both conditions.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

The purpose of the study is to construct and test a ‘course recommendation system’ which is developed by conducting the CBR approach. The main aim of applying the CBR approach in developing the system is to provide course suggestion for the students. With the object-oriented structure of the developed software, the sufficiency can be ensured while there is a need for reconfiguration. Thus, there is the possibility of expanding it in any kinds of improvements. When the system is in the coding process, MVC- based coding is used, because it is easy to use and is also comprehensible for large-scale systems.

The result of this study has showed that the Web-based project had reached success. The related literature states that students need to be guided as making their course selections [37]. Thus, the "Course Suggestion System" Web page was designed by such a need. It is aimed that students can find all kinds of information on a single web portal. Even though this site is designed to satisfy the request, the needs of each student are hardly meet. This web-based project is tested by students who had just made course selection without the grades of the midterm and final exams. The given data is compared to see whether the taken courses are compatible with the suggestion system. We have learned the ratio of success while making this system. Many studies test their system and use CBR method in their systems.

According to the study in [3], a website based on Natural Language processing was created for the stage of presenting the new experiences to the user by benefiting from the past similarities. The case-based reasoning was used in this study. Besides, Scrum-based progress was recorded in this study. The lowest level of similarity was determined to be 50 percent. The case-base of this project is quite small. The values

to be realized are in the range of 0.8 to 1. The system was tested by considering the values in this range. As a result, the feedback rate from certain people to learn the target outcome is 57 percent.

In [38], the case-based reasoning method was used to provide the most appropriate solution by using similar past experiences while making the diagnosis for people with anxiety. In this article, similarity ratios are calculated by using the nearest algorithm. The NN algorithm was used in this study to obtain similarities between the two closest neighbors by looking at the data. As a result, the success rate of the cases obtained is 65 percent.

In [39], a learning algorithm was created based on the data in the database. This is done using case-based reasoning. Based on the information in the past, the system is tested by using the simulation application to the users. In this study, 0-1 range is based on as similarity ratio range. The found similarity range has been accepted to be within these ranges. As a result, Auto Learning in this study was successful in better learning of users.

In [40], a computer-based information system was created. This system is designed for Microelectromechanical systems. Information in the past has been used for the solutions of the designs. The case-based reasoning was used in this study. The primary purpose of the system made in this study is to re-update the information in the existing database according to the new problems to the designer or to use the new information. A data library has been created for this system. As a result, in this study, which focuses on the resonators provided in the test of this system, the result for this section is 82%. According to the result of the system test, calculate each recommendation points for 100 people and our test result is %68. The success rate of the system was calculated by calculating the average of the highest similarity rates for the test students. This test result is % 72. The system achieved in both test areas.

The results of this study are compatible with the results of the previous studies. We found similar success rate with previous studies that presents the results in other domains such as health sector. This thesis aims to guide the users with the developing

Web-based recommendation system with the suggestions generating by the CBR system. In this sense, studies have been carried out as a further progressive basis. By making improvements and additions, it is aimed to make our system more suitable for the user's wishes in the future. For now, it is intended from the system to provide course-based recommendations and to make course selections according to the success of the courses besides the progress of the students.

REFERENCES

[1] J. Kolodner, Case-based reasoning, San Mateo, 2014. Morgan Kaufmann

[2] J. A. Recio-GarcíaDerek, BridgeBelén, D.-A. A. and González-Calero, "CBR for CBR: A Case-Based Template Recommender System for Building Case-Based Systems," *In European Conference on Cased-Based Reasoning*, pp. 459-473, 2008.

[3] R. G. Rocha, "A case-based reasoning system to support the global software development," *Procedia Computer Science*, vol. 35, pp. 194-202, 2014.

[4] J. Nordlund and H. Schafer, "Case-Based Reasoning in a Support System," *Master's thesis, UMEA University, Sweden*, 19 April 2006.

[5] R. L. de Mántaras and E. Plaza, "Case-Based Reasoning: an overview," *Al communications*, pp. 21-29, 1997.

[6] J. Castro, M. Navarro, J. Sánchez and J. Zurita, "Loss and gain functions for CBR retrieval," *Information Sciences*, pp. 1738-1750, 13 May 2009.

[7] P. Cunningham and A. Bonzano, "Knowledge engineering issues in developing a case-based reasoning application," *Knowledge-Based Systems*, pp. 371-379, November 1999.

[8] J. L. Kolodner and M. Y. Jona, Case-Based Reasoning: An Overview, Defense Advanced Research Projects Agency, November 1, 2000.

[9] M. M. Richter and R. O. Weber, Case-Based Reasoning, Springer-Verlag Berlin An, 2016.

[10] S. Begum, S. Barua and M. U. Ahmed, "Physiological Sensor Signals Classification for Healthcare Using Sensor Data Fusion and Case-Based Reasoning," *Sensors*, pp. 11770-11785, 3 July 2014.

[11] J. Main, T. S. Dillon and S. C. K. Shiu, "A Tutorial on Case Based Reasoning," *Soft Computing in Case Based Reasoning*, pp. 1-28, 2001.

[12] R. L. D. MANTARAS, D. MCSHERRY, D. BRIDGE, D. LEAKE and ..., "Retrieval, reuse, revision and retention in case-based reasoning," *The Knowledge Engineering Review,* pp. 215-240, 08 May 2006.

[13] J. L. Kolodner, "The role of experience in common sense and expert problem solving," *Georgia Institute of Technology,* 1985.

[14] S. Agrawal and R. D. Gupta, "Development and comparison of open source based Web GIS Frameworks on WAMP and Apache Tomcat Web Servers," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences,* 2014.

[15] J. L. Kolodner, "An introduction to case-based reasoning," *Artificial intelligence review,* pp. 3-34, 1992.

[16] F. Lorenzi and F. Ricci, "Case-based recommender systems: A unifying view," *In Intelligent Techniques for Web Personalization,* pp. 89-113.

[17] M. Montaner, Collaborative Recommender Agents Based on Case- Based Reasoning and Trust, 2003.

[18] A. M. Islam, "Case Based Reasoning method for analysis of Physiological sensor data.," 2012.

[19] J.B.Noh, K. Lee, J. Kim, J. Lee and S. Kim, "A case-based reasoning approach to cognitive map-driven tacitknowledge management," pp. 249-259, 2000.

[20] BelénDíaz-Agudo, P. A. González-Calero, J. A. Recio-García and A. A. Sánchez-Ruiz-Granados, "Building CBR systems with jcolibri," pp. 68-75, December 2007.

[21] P. Jackson, K. Al-Kofahi, A. Tyrrell and A. Vachher, "Information extraction from case law and retrieval of prior cases," *Artifical Intelligence,* pp. 239-290, November 2003.

[22] A. D. Renzis, M. Garriga, A. Flores, A. Cechich and A. Zunino, "Case-based Reasoning for Web Service Discovery and Selectio," *Electronic Notes in Theoretical Computer Science,* pp. 89-112, March 2016.

[23] R. Pan, Q. Yang and S. J. Pan, "Mining competent case bases for case-based reasoning," *Artifical Intelligence,* pp. 1039-1068, 30 April 2007.

[24] V. Aleven, "Using background knowledge in case-based legal reasoning: A computational model and an intelligent learning environment," *Artificial Intelligence,* pp. 183-237, November 2003.

[25] S. Lee and K.-j. Kim, "Using case-based reasoning for the design of controls for internet-based information systems," *Expert Systems with Applications*, pp. 5582-5591, April 2009.

[26] M. M. Richter, "The search for knowledge, contexts, and Case-Based Reasoning," *Engineering Applications of Artificial Intelligence*, pp. 3-9, February 2009 2009.

[27] I. Watson, "Case-Based Reasoning is a Methodology not a Technology," *In Research and Development in Expert Systems XV*, pp. 213-223, 17 March 1999.

[28] A. K. Çınar, *Integrating case based reasoning and geographic information systems in a planing support system: Çeşme Peninsula study*, 2009.

[29] A. Holt, S. G. MacDonell and G. L. Benwell, "Assessing the degree of spatial isomorphism for exploratory spatial analysis," *GeoComputation*, 1998.

[30] E. Enginer, *PHP, SOL ve AJAX teknolojileri kullanarak Hacettepe Üniversitesi biyoloji bölümü herbaryumu için bir sorgulama yazılımının geliştirilmesi.,* MS thesis. Başkent Üniversitesi Fen Bilimleri Enstitüsü, 2009.

[31] R. Nixon, Learning PHP, MySQL, JavaScript, and CSS: A step-by-step guide to creating dynamic websites, " O'Reilly Media, Inc., 2012.

[32] A. Leff and J. T. Rayfield, "Web-application development using the model/view/controller design pattern.," *Enterprise Distributed Object Computing Conference*, pp. 118-127, 2001.

[33] Q. Ni, A. B. G. Hernando and I. P. d. l. Cruz, "The Elderly's Independent Living in Smart Homes: A Characterization of Activities and Sensing Infrastructure Survey to Facilitate Services Development," 14 May 2015.

[34] J. Deacon, "Model-view-controller (mvc) architecture," April 2005. [Online]. Available: http://www.jdl.co.uk/briefings/index.html#mvc .

[35] G. E. Krasner and S. T. Pope, "A description of the model-view-controller user interface paradigm in the smalltalk-80 system," *Journal of object oriented programming* , vol. 1.3, no. 26-49, 1988.

[36] D. B. Leake, "CBR in context: The present and future. Case-based reasoning," *Experiances,lessons, and future directions*, pp. 3-30, 1996.

[37] D. Hu, *CASE-BASED REASONING (CBR) FOR CHILDREN STORY SELECTION*, 2003.

[38] R. Janssen, P. Spronck and A. Arntz, "Case-based reasoning for predicting the success of therapy," *Expert Systems*, no. 32.2, pp. 165-177, 2015.

[39] J. Sebestyénová, *Case-based Reasoning in Agent-based Decision Support System*, Slovakia, 2007.

[40] C. L. Cobb and A. M. Agogino, "Case-based reasoning for evolutionary MEMS design," *Journal of Computing and Information Science in Engineering*, 2010.

[41] R. C. Schank, "Dynamic memory: A theory of reminding and learning in computers and people," *Cambridge: Cambridge University Press*, 1982.